AFIT/GCA/LAS/97S-1

CALIBRATION AND VALIDATION OF THE COCOMO II.1997.0 COST/SCHEDULE ESTIMATING MODEL TO THE SPACE AND MISSILE SYSTEMS CENTER DATABASE

THESIS

Wayne A. Bernheisel
1st Lieutenant, USAF

AFIT/GCA/LAS/97S-1

Approved for public release; distribution unlimited

AFIT/GCA/LAS/97S-1

CALIBRATION AND VALIDATION OF THE COCOMO II.1997.0 COST/SCHEDULE ESTIMATING MODEL TO THE SPACE AND MISSILE SYSTEMS CENTER DATABASE

THESIS

Presented to the Faculty of the Graduate School of Logistics

and Acquisition Management of the Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Cost Analysis

Wayne A. Bernheisel, B.S.

1st Lieutenant, USAF

September 1997

Preface

I first and foremost would like to thank God for giving me the perseverance, capability, and skill necessary to complete this program! Without God, none of this would have been possible.

This program has routinely necessitated long, isolated hours of work, and therefore, required great sacrifice. I deeply appreciate my wife, Jennifer, and kids, Ashley and Kristen, for supporting me during these times. The time lost with my family can never be recovered, but their support and love have been invaluable to me.

I would like to thank Professor Ferens, my advisor, and Dr. Christensen, my reader, for their guidance and dedication to helping me produce a quality research effort and for their concern for me as a student. Their thoroughness, knowledge, and help has not gone unnoticed, and is sincerely appreciated.

Special thanks to Colonel Ed Weeks, Mr. Dave Hansen, and Ms. Sherry Stukes. Colonel Weeks, the Director of the Air Force Cost Analysis Agency, supplied the funds which allowed me the opportunity to go TDY to meet and discuss the COCOMO II model with Dr. Boehm in Crystal City, VA. Mr. Hansen, the Chief of Financial Management at the Space and Missile Systems Center (SMC), acted as my sponsor and Ms. Stukes of MCR Corporation supplied me with the SMC Software Database and offered support for its use.

Overall, I believe that my AFIT education and experience have been a great opportunity well worth the time and effort. The Cost Analysis program is a high caliber, cutting edge program, which is due to the efforts of individuals like Lieutenant Colonel Giuliano and Dr. Roland Kankey; thank you both for your dedication, concern, and integrity.

Any inquiries about this database can be directed to: Sherry Stukes, MCR Federal, Inc., (805) 496-7111, sstukes@mcri.com or Shirley Tinkler, SMC/FMC, (310) 363-5057, DSN 833-5057, tinklersm@post5.laafb.af.mil.

Wayne Bernheisel

Table of Contents

Page

Preface ii

List of Figures vi

List of Tables vii

Abstract ix

I. Introduction 1

Overview 1
General Issue 2
Specific Issue 2
Research Objectives 3
Scope of Research 4
Thesis Overview 5

II. Literature Review 6

Overview 6
Previous Studies 6
Analysis of AFIT Studies 6
Analysis of Other Studies 10
Why Are Software Cost Estimating Models Inaccurate? 12
Overview of Literature 12
Air Force Viewpoint 14
Industry Viewpoint 19
COCOMO II 30
Overview 30
Comparison of COCOMO 1981 TO COCOMO II 30
Model Weaknesses 33
Model Equations 35
Summary 36

III. Methodology 37

Overview 37
Procedures and Data Analysis 37
SMC SWDB 37
SMC SWDB Query Setup 38

Page

Performing the Query 40
SMC SWDB Weaknesses Identified 41
Model Calibration 43
Resampling Method 44
Analysis Methodology 45
Regression Analysis 47
Summary 51

IV. Findings 53

Overview 53
SMC SWDB Stratification 53
Analysis of SMC SWDB Fields to COCOMO II 55
Calibrating the Coefficient 63
Calibration & Validation Results 64
Regression Analysis 73
Summary 80

V. Conclusions and Recommendations 81

Overview 81
Conclusions 84
Recommendations 86

Appendix A. Acronyms and Glossary of Terms 89

Appendix B. SMC SWDB Data Sets 92

Appendix C. Project Selection for Calibration and Validation 102

Appendix C1. Coefficient Calibration 104

Appendix C2. Default and Calibration Estimates 116

Appendix D. Regression 118

References 124

Vita 130

List of Figures

Figure Page

1. Regression Variables 48
2. Default Estimate vs. Actual Effort—Military Ground, C^2 65
3. Default Estimate vs. Actual Effort—Military Ground, Signal Processing 65
4. Default Estimate vs. Actual Effort—Ground in Support of Space 66
5. Default Estimate vs. Actual Effort—Military Mobile 66
6. Default Regression Equations 74
7. SAS Program 76
8. Final Regression Models 79

List of Tables

Table Page

1. Summary of AFIT Calibration/Validation Efforts 7
2. Causes by Responsibility for Inaccurate Estimates 15
3. Correlation with Inaccuracy Percentage 15
4. COCOMO Input Parameter Comparison 32
5. SMC SWDB Queries 39
6. SMC SWDB Software Phase Normalization 42
7. Calibrating the Coefficient *A* 44
8. SMC SWDB Resultant Queries 55
9. Reuse Requirements 57
10. Team Language and Tool Experience 60
11. USER 2 - Development Volatility Rating 61
12. Range of 'A' Coefficient 63
13. Military Ground--C^2, Range Percentages of Effort 70
14. Military Ground—Signal Processing, Range Percentages of Effort 70
15. Ground in Support of Space, Range Percentages of Effort 70
16. Military Mobile, Range Percentages of Effort 71
17. Wilcoxon Test Results 72
18. Results of Default Model Accuracy 73

Table | Page

19. Results of Calibrated Model Accuracy 73

20. Initial Regression Run Results 74

21. Final Regression Run Results 78

22. Accuracy Results of Final Regression Model 79

23. Comparison of the Calibration Data Set Criteria 82

24. Comparison of the Validation Data Set Criteria 82

25. Accuracy Results of Final Regression Model 83

AFIT/GCA/LAS/97S-1

Abstract

The pressure to decrease costs within the Department of Defense has influenced the start of many cost estimating studies, in an effort to provide more accurate estimating and reduce costs. The goal of this study was to determine the accuracy of COCOMO II.1997.0, a software cost and schedule estimating model, using Magnitude of Relative Error, Mean Magnitude of Relative Error, Relative Root Mean Square, and a 25 percent Prediction Level. Effort estimates were completed using the model in default and in calibrated mode. Calibration was accomplished by dividing four stratified data sets into two random validation and calibration data sets using five times resampling.

The accuracy results were poor; the best having an accuracy of only .3332 within 40 percent of the time in calibrated mode. It was found that homogeneous data is the key to producing the best results, and the model typically underestimates. The second part of this thesis was to try and improve upon the default mode estimates. This was accomplished by regressing the model estimates to the actual effort. Each original regression equation was transformed and tested for normality, equal variance, and significance. Overall, the results were promising; regression improved the accuracy in three of the four cases, the best having an accuracy of .2059 within 75 percent of the time.

CALIBRATION AND VALIDATION OF THE COCOMO II.1997.0 COST/SCHEDULE ESTIMATING MODEL TO THE SPACE AND MISSILE SYSTEMS CENTER SOFTWARE DATABASE

I. Introduction

Overview

Over the last two decades, we have seen a growing trend to use software cost models to estimate the cost of developing software. These models allow us to estimate costs and schedules more quickly and easily than using traditional methods. To date though, there has been no proof that shows software cost models to be consistently accurate within 25% of the actual cost, 75% of the time (based on Conte's criteria), except for CHECKPOINT (Ferens, 1997).

One of the first software cost models to be developed was the "Nelson" model in 1965 (Ferens, 1997). Since this time, we have observed many modifications, updates, and introductions of new models, which total approximately 50 models in the United States (Jones, 1996:19). A common modification among most models has been to increase the number of input parameters. Some models have been inundated with inputs and output features, yet the accuracy of these models has shown little improvement. Although a great amount of research, time, and money has been devoted to improving our situation, other factors in software development, software complexity, standardization, and lack of data greatly inhibits the ability of software cost model designers to develop credible models.

General Issue

Everyone is aware of the pressures to decrease Federal spending. Unfortunately, the military is funded using discretionary funding, and when there is little perceived threat to national security, the military funding is targeted for reduction. In fact, the real rate of military funding has decreased every year since 1985 (D'Angelo, 1997). Public scrutiny and awareness of defense spending increased when the media released information that the services spend exorbitant amounts of money for hammers, toilet seats, and other common-use items. This scrutiny has been amplified with growing concern over the Federal budget deficit and the lack of a notable threat to our way of life.

As cost analysts, our job is to perform the most accurate cost, schedule, and risk analysis of projects so that program managers may make informed decisions. But, in estimating software costs, our ability to provide accurate estimates early in a program's development is extremely limited. The current status of our situation was best summed up in a 1994 speech by Lloyd K. Mosemann, II, the Deputy Assistant Secretary of the Air Force (Communications, Computers, and Support Systems). An excerpt from this speech follows.

> From a Pentagon perspective, it is not the fact that software costs are growing annually and consuming more and more of our defense dollars that worries us. Nor is it the fact that our weapon systems and commanders are becoming more and more reliant on software to perform their mission. Our inability to predict how much a software system will cost, when it will be operational, and whether or not it will satisfy user requirements, is the major concern. What our senior managers and DOD (Department of Defense) leaders want most from us, is to deliver on our promises. They want systems that are on-time, within budget, that satisfy user requirements, and are reliable. (Mosemann, 1994)

Specific Issue

There are several well known software cost model experts in the United States, but few, if any, with the reputation and credibility of Dr. Barry Boehm, Professor of

Computer Science, University of Southern California (USC). For several years, Dr. Boehm has been advising and directing USC graduate students at both the masters and doctorate level, in developing and performing supportable research in the development of the much anticipated and long awaited COCOMO II.1997.0 (Constructive Cost Model, 1997 model, version 0—referred to as just COCOMO II throughout the rest of the text). This is the updated version of the original COCOMO which was released by Dr. Boehm in 1981 (Boehm, 1981). COCOMO has probably been the most utilized of all software cost estimating models when all the subsequent versions like COCOMO-R, Ada COCOMO, and REVIC, just to name a few are considered (Boehm, presentation 1997).

The purpose of this research is to calibrate and determine the accuracy of the COCOMO II model to the Air Force Space and Missile Systems Center (SMC) Software Database (SWDB) that was created by Management Consulting and Research, Inc. (MCR) (SMC SWDB, 1995).

Research Objectives

This effort will be focused on calibrating the effort equation coefficient of the COCOMO II Software Cost and Schedule Model in the Post-Architecture mode to specific applications (i.e. Military Ground, Avionics, Unmanned Space) within the SMC Database, Version 2.1. The purpose of the calibration is to determine the accuracy (goodness of fit) of the model in default (uncalibrated) and calibrated modes, and validate the model's use by SMC and other DOD agencies to estimate program costs and schedules. The following criteria, as determined by Conte, Dunsmore, and Shen in their book Software Engineering Metrics and Models, will be used to evaluate and validate the accuracy of the estimates: Mean Magnitude of Relative Error (MMRE) less than 0.25, Relative Root Mean Square (RRMS) less than 0.25, and Prediction Level (Pred) of 0.25 in 75% of the time (Conte, Dunsmore, & Shen, 1986:172-175).

The research questions to be answered include:

1. What is the uncalibrated accuracy of the COCOMO II model in the Post-Architecture mode when estimating efforts in the SMC SWDB for both the calibration and validation subsets?

2. What is the calibrated accuracy of the COCOMO II model in the Post-Architecture mode when estimating efforts in the SMC SWDB for both the calibration and validation subsets?

3. Are there any improvements in accuracy between the calibrated and uncalibrated settings of the COCOMO II model in the Post-Architecture mode?

4. In its current form, is the COCOMO II model useful for DOD cost analysts on software development projects?

Scope of Research

This effort is restricted to the calibration and validation of the COCOMO II model to the SMC SWDB, Version 2.1. The results of this research effort are limited by the accuracy and validity of the contractor data recorded in the SMC SWDB. The study will not include an analysis of project risk, schedule allocations, and support/maintenance, or if released prior to the completion of this thesis, an evaluation of a later version of COCOMO II (COCOMO II.1997.1 is due to be released in the Summer, 1997). Up to this time, there are no known published calibrations by independent researchers, including the Air Force, of the 1981 COCOMO model. There was a study done by MCR in 1994 and two done by previous Air Force Institute of Technology (AFIT) Master's students (Ourada 1991 and Weber 1995) on the calibration of the Revised and Enhanced Version of Intermediate COCOMO (REVIC), the Air Force version of COCOMO developed by Kile. When applicable, the results of this effort will be compared to the results of these two previous theses to try and determine if there are any identifiable improvements.

Thesis Overview

This effort will use the SMC SWDB, Version 2.1 to calibrate COCOMO II.1997.0. A thorough investigation of the literature relevant to this thesis will be summarized in Chapter II. The modifications made to the original COCOMO will be identified as well as explanations for the changes and/or additions. Chapter II will also include efforts that support and oppose the COCOMO II methodology as well as a history of all known previous and current research in which software cost and schedule models were calibrated to DOD and non-DOD environments. The methodology and steps taken in this study will be discussed at such detail in Chapter III to permit replication of this study. This will include the method used in the calibration and validation of the COCOMO II model, and stratification of the SMC SWDB. Results of the calibration, validation, and any further noted limitations and strengths of the model will be presented and assessed in Chapter IV. Lastly, Chapter V will encompass any recommendations for future research and any significant findings felt necessary to restate or add. The Appendices at the end of the thesis contain a glossary of acronyms and technical terms, the data used in the analysis, a comparison chart of COCOMO and COCOMO II and detailed tables and spreadsheets of computations.

The desire is that the COCOMO II cost and schedule software model will provide accurate estimates based on the criteria set forth. This will then provide Air Force SMC and other cost analysts with a credible (and calibrated) software cost and schedule model to develop program estimates. In turn, the analysts can then provide program managers with accurate software costs and schedules to base and, hopefully, optimize their decisions.

II. Literature Review

Overview

The purpose of this chapter is for the reader to become current on software cost estimating issues by discussing results of previous similar studies and to identify some of the reasons for model inaccuracies. This will be followed by an in-depth analysis of the COCOMO II model, any revisions and methodology changes from the 1981 COCOMO version, and lastly, by an analysis of the SMC SWDB. To begin, it's important to understand the situation from the Air Force perspective.

> The software industry is reaching its 50 year mark, however, the same problems that plagued us 20 years ago still persist. DOD has had a distressing history of procuring elaborate, high-tech software-intensive weapons that do not work, cannot be relied upon, modified, or maintained.... With virtually every acquisition snafu, the software component can be isolated as the prime source of our dilemmas. (Department of the Air Force (DAF), 1996:Sec 1, 1)

Previous Studies

This section will focus on published software cost model studies beginning with an analysis of nine prior AFIT studies and ending with an analysis of three other (non-AFIT) studies. The intent is not to criticize the previous studies, but to learn from them by analyzing their results and methodology. This information can then be used to strengthen the methodology, consistency, and results of this study.

Analysis of AFIT Studies. In Table 1, Summary of AFIT Calibration/Validation Efforts, on the following page, is a breakdown of eight of the nine AFIT studies conducted from 1990 to 1996. The Daly thesis (the ninth AFIT thesis) will be discussed later since it was not consistent with how the other theses were analyzed.

Table 1. Summary of AFIT Calibration/Validation Efforts

Study	Cost Model	Application Type	Cal.	Val.	Default Accuracy			Validated Accuracy		
					MMRE	RRMS	Pred (0.25)	MMRE	RRMS	Pred (0.25)
Ourada (91)	REVIC	Mil Grnd	X	X	n/r	n/r	0.57	n/r	n/r	0.28
Galonsky (95)	PRICE-S	Mil Grnd	X	X	not reported		0.52	not reported		0.48
		Unmnd Space	X	X	not reported		0.36	not reported		0.50
		Missile	X		not reported		0.75	not reported		0.75
		Mil Mobile	X		not reported		0.38	not reported		0.38
Kressin (95)	SLIM	Mil Grnd - MIS	X		0.962	n/r	0.00	0.157	n/r	0.83
		Mil Grnd - All	X		n/r	n/r	n/r	2.166	n/r	0.08
		C^2	X	X	0.621	n/r	0.00	0.666	n/r	0.00
Rathmann (95)	SEER-SEM	Avionics	X	X	0.923	1.472	0.00	0.243	0.240	1.00
		C^2	X	X	0.531	1.031	0.43	0.311	0.296	0.29
		Signal Proc	X	X	1.440	1.082	0.29	2.092	1.610	0.43
		Mil Mobile	X	X	2.802	3.711	0.11	0.462	0.342	0.25
Vegas (95)	SASET	Mil Grnd	X	X	10.04	n/r	0.00	5.820	n/r	0.38
		Unmnd Space	X	X	5.54	n/r	0.23	0.940	n/r	0.00
		Avionics	X	X	1.760	n/r	0.00	0.220	n/r	1.00
		Mil Mobile	X	X	5.610	n/r	0.25	3.570	n/r	0.00
Weber (95)	REVIC	Mil Grnd	X	X	1.21	1.13	0.00	0.86	0.68	0.00
		Unmnd Space	X	X	0.44	0.62	0.50	0.32	0.34	0.50
Mertes (96)	CHECKPOINT	MIS - COBOL		X	0.542	0.101	0.67	0.018	0.010	1.00
	(f.p.)	Mil Mobile-Ada		X	1.384	0.412	0.25	0.192	0.057	0.75
	(f.p.)	Avionics		X	0.817	0.685	0.50	0.158	0.111	0.75
	(sloc)	C^2		X	0.193	0.145	0.50	0.165	0.156	0.50
	(sloc)	Signal Proc		X	0.090	0.081	1.00	0.090	0.081	1.00
	(sloc)	Unmnd Space		X	0.048	0.050	1.00	0.040	0.055	1.00
	(sloc)	Grnd-spt. space		X	0.050	0.058	1.00	0.050	0.058	1.00
	(sloc)	COBOL Projs		X	0.050	0.051	1.00	0.049	0.051	1.00
Southwell (96)	SOFTCOST-R	Mil Grnd	X	X	1.895	3.433	0.00	0.519	0.870	0.83
		Signal Proc	X	X	0.430	0.612	0.11	0.282	0.634	0.44
		Unmnd Space	X	X	0.557	1.048	0.20	0.480	0.923	0.20
		Grnd-spt. space	X	X	2.734	3.125	0.13	1.802	1.966	0.20
		Mil Mobile	X	X	0.635	0.514	0.20	0.420	0.395	0.40
		Avionics	X	X	0.713	0.758	0.20	0.846	0.568	0.20

Table 1 is the result of a collaborative effort of the author and two other AFIT students, Dave Marzo and Tom Shrum, and the information provided within the table is obtained directly from the respective theses (Marzo, 1997 and Shrum, 1997). Table 1 includes author name, cost model name, application type, whether calibrated and/or validated, default accuracy (MMRE, RRMS, Pred, K/N), and validated accuracy (MMRE, RRMS, Pred, K/N). K/N is the percentage of estimates that fall within the specified prediction level of 25 or 30 percent. It's important to note that the Ourada thesis was actually based on a 30 percent prediction level versus a 25 percent prediction level. Each one of these studies is available through the AFIT library or through DTIC (Defense Technical Information Center).

Up to this point, the AFIT research has been geared towards the most regularly used Air Force and new software cost models. The objective has been to determine the accuracy of each model applied to varying military applications. A weakness of each analysis was the lack of usable historical data. In most cases, the researchers found a gold mine if they had more than 12 data points. The norm appears to be less than 10 data points.

A second weakness of these studies is inconsistency. Half of the studies did not validate and/or report all their findings. The CHECKPOINT model was calibrated, but the calibrated model was not validated using the calibrated data sets. This inconsistency and oversight in the methodology with the studies has been identified and stimulated greater consistency in the latter studies. In fact, there are currently two other studies being conducted at AFIT. The SAGE model is being calibrated and validated by Marzo to the SMC SWDB and Electronic Systems Center (ESC) Database (Marzo, 1997), and CHECKPOINT is being calibrated and validated by Shrum to the ESC Database (Shrum, 1997). Each of these models, including COCOMO II will report all RRMS and MMRE values as well as validate the models using a similar methodology.

A third weakness of the studies is the validation technique used. Except for the Galonsky study (Galonsky, 1995), the validation for the 1995 studies were accomplished using the following steps:

1) If total data points ≤ 8, use all points for calibration.
2) If total data points $\geq 9 \leq 12$, use 8 points for calibration and the remaining for validation.
3) If total data points > 12, use 2/3 of the points to calibrate and the remaining points for validation.

The 1996 AFIT theses differed in validation technique in that they validated the models using a 50/50 method. The students used half of the data points for calibration and the other half for validation. Although both of these methods are valid and give sound results, there are more robust techniques better accepted by the technical community.

One such method that is suggested by Clark, a PhD student at USC working with Dr. Boehm on the COCOMO II development, is to randomly calibrate 80 percent of the data points and project the remaining 20 percent, repeating this procedure five times (Clark, 1997). Some may recognize this technique as the resampling method. This method is valuable because it enhances the credibility to studies done with fewer than a fundamentally robust set of data points. In the Galonsky study, a variation similar to this method was conducted, and lends itself to a similar robustness (Galonsky, 1995).

Even though there have been these shortcomings, the most significant findings of the previous studies lies in their results. Overwhelmingly, the results show improvement from uncalibrated to calibrated results. This shows that the models, when calibrated to the environment, provide more accurate estimates, and reinforce the need for accurate, and consistent historical data to calibrate the models. Except for the study done using CHECKPOINT, no model was consistently accurate. The two studies that did report a 100 percent accuracy under the validation heading, used only one data point. The use of only one data point (without using resampling) further solidifies some of the inconsistencies in the studies. Due to the remarkable results achieved by the CHECKPOINT model even without being calibrated, this model will also be analyzed using the Electronic Systems Center (ESC) Database to further validate its significant findings. Results of this study should be completed by September 1997 (Shrum, 1997).

The last AFIT thesis to be discussed is that done by Daly. In this 1990 study, Daly chose five models (REVIC, PRICE-S, SEER, System-4, and SPQR/20) to estimate schedule for 21 separate projects from the Electronic System Division (now ESC). After he computed the estimates with the models, he regressed the estimates against the actual schedule values to determine a goodness of fit, R^2. Daly found no model by itself was accurate within 30 percent of the actual schedule, 70 percent of the time (Daly, 1990). After regressing the estimates for each model, he found that only the System-4 seemed to be consistent in its estimates versus the actual schedules (Daly, 1990:59). This implies that an analyst could run the System-4 model and then use the estimate in a regression equation to determine a more accurate schedule estimate. Daly then found System-4 to be accurate within 30 percent, 71.4 percent of the time (Daly, 1990:85).

Analysis of Other Studies. There are three other research efforts that have been published that add insight to this study. These efforts were chosen because of availability and due to a significant result or methodology utilized.

Kemerer Study. The first and most significant was that done by Kemerer from Carnegie-Mellon University. Kemerer validated four models (SLIM, 1981 COCOMO, Albrecht's Function Points, and ESTIMATICS) using 15 business data points (except for ESTIMATICS, which was only validated using 9) in the uncalibrated mode. The results were not surprising. ESTIMATICS and the Albrecht's Function Points model outperformed the SLIM and COCOMO model, since the latter two models were developed using DOD projects (Kemerer, 1987). Like CHECKPOINT, both the

ESTIMATICS and the Albrecht's Function Points models use Function Points (FPs) within the algorithms to determine effort instead of converting FPs to SLOC (Source Lines of Code).

The significance of this article is not the findings, but Kemerer's use of regression to determine patterns and improve accuracy. After computing estimated manmonths, Kemerer regressed each estimate to the actual manmonths. A high R^2 indicated a consistency of the model to over or under estimate a project, similar to that done in the 1990 Daly AFIT thesis (Kemerer, 1987). The SLIM model had the highest R^2 of 87.8 percent (Kemerer, 1987:422). The real fascination with this test was with COCOMO. Kemerer analyzed the model in all three modes (Basic, Intermediate, and Detailed) and found as the model became more detailed, the lower the R^2. This suggests that the added parameters are not contributing to the overall effectiveness of the estimate (Kemerer, 1987:422-423). However, the main weakness with Kemerer's methodology is that he failed to test the assumptions of each regression equation, nevertheless, the idea to use regression to improve model estimate accuracy is significant (Matson, Barret, and Mellichamp, 1994:278-280).

<u>Thibodeau and IITRI Studies</u>. The other two studies to review include a study done by Thibodeau in 1981 and a study done by IIT Research Institute (IITRI) in 1989. In the Thibodeau study, he calibrated nine models using three databases (Thibodeau, 1981). The significance of this study are as follows:

1) Results greatly improved with calibration, in fact, as high as a factor of five (Thibodeau, 1981:5-29).
2) Models consistently obtained better results when used with certain types of applications (Thibodeau, 1981).

The IITRI study was significant because it analyzed the results of seven cost models (PRICE-S, two variants of COCOMO, System-3, SPQR/20, SASET, SoftCost-Ada) to eight Ada specific programs. Ada was specifically designed for and is the principal language used in military applications, and more specifically, weapons system software (Ferens, 1997). Weapons system software is different then the normal corporate type of software, commonly known as Management Information System (MIS) software (Ferens, 1997). The major differences between weapons system and MIS software, are that weapons system software is real time and uses a high proportion of complex mathematical coding (Ferens, 1997). Up to 1997, DOD mandated Ada as the required language to be used unless a waiver was approved. Lloyd Mosemann stated:

> Even as DOD moves from mandating Ada to preferring Ada, any company would be foolish to establish a product-line based on any other language now known. The special features of Ada, such as tasking and exception handling, make it mandatory for any application involving safety of life.... (Department of the Air Force, 1996:iii)

The results of this study, like other studies, showed estimating accuracy improved with calibration. The best results were achieved by PRICE-S and System-3 (predecessor to SEER-SEM). Both models were accurate within 30 percent, 62 percent of the time. The IITRI study, as well as the Thibodeau study, did not use validation techniques.

Why Are Software Cost Estimating Models Inaccurate?

Overview of Literature. One obvious point that can be made when examining the incredible amounts of literature available concerning software engineering and why software is seldom on time or within budget, is that no one has proved their view is the correct one. If so, there would be a proven cost model that consistently produced accurate estimates, even though there is a overwhelming plethora of ideas. Therefore, it appears that software cost estimating is similar to predicting the weather. There are an infinite number of factors involved, open to numerous interpretations, which may result in one of nine outcomes involving schedule and budget (i.e. within schedule, but cost higher than budgeted, etc.). By summarizing previous studies, we know that:

1) Calibration usually improves accuracy;
2) Models seem to be able to estimate more accurately within certain applications; and,
3) The user needs to become as familiar as possible with the chosen model to understand its weaknesses, strengths, and sensitivities (Ferens, 1997).

If we can improve estimates of cost and schedule by following these three points, the odds are still against us that during the early stages of a program; we will not be able to produce an estimate we are confident in. Dr. Boehm reported that during the Feasibility phase of software development, an estimate could be off by as much as a factor of four (Boehm, 1981:311). His findings further show that the knowledge and understanding required of the development is not known well enough to produce an accurate estimate (based on Conte's criteria) until the Product Design phase and later (Boehm, 1981:311). This is quite distressing because the software is being coded at this point. Therefore, the question must be asked, "Can the models be changed, modified, or updated to produce more accurate results, or are there some other factors involved that are making the estimates look bad"? One reason for inaccurate estimates may be due to assumptions.

> One of the commonest methods in the programming industry for expressing the relative costs of programming activities is the use of percentages or ratios, such as the historical rule of thumb for assembler language programs that design will take 20 percent of a software development cycle, coding will take 30 percent, (and) integration and testing will take 50 percent.... The first problem with using percentages is that they break down completely when programs in different languages are being compared. (Jones, 1986:13)

These percentages may also be affected by multisite development, tools that are new or insufficient, and programmer and analyst experience (Jones, 1986:13). From the perspective of the cost estimating model developers, it's probably safe to say that they feel it's some other factor involved and not the models themselves. From the perspective of the software developers, they may feel it's the model. Lastly, from the perspective of a cost analyst, it's probably due to an overoptimistic input, requirements change, budget cut, or miscommunication.

Lederer and Prasad Study. A 1995 study utilizing questionnaire data from 112 different private organizations reported the causes of inaccurate software development costs. The target audience for the questionnaire was information system managers and professionals. The possible causes of inaccurate estimates are recreated and listed in Table 2 on the following page, with the most common response listed first. The research by Lederer and Prasad initially found that the user may be at the forefront of the problem. However, with persistent investigation, the researchers found it quite the opposite. Lederer and Prasad classified the causes into four categories. These categories were then correlated with actual inaccurate estimates within the respondents' organizations. The results are listed in Table 3 on the following page. Lederer and Prasad found that the "...information systems managers and professionals greatly attribute inaccurate estimates to users and poor communication with them (as seen in Table 2); but, in fact, project control may be more responsible. This finding implies that information systems managers and professionals may want to reevaluate their attitudes toward their users (Lederer and Prasad, 1995:132-133)". Although this study is based on information systems software development, it does identify some common issues that also plague the military environment.

Air Force Viewpoint. One of the issues concerning software development is whether it is an art or science? The answer to this creates difficulties because, if it is an art, then the institutionalization of the development process is most likely the wrong approach. However, if it is a science, then guidelines, metrics, and direction are

Table 2. Causes by Responsibility for Inaccurate Estimates

Causes
Change requests by users
Users lack of understanding of their own requirements
Overlooked tasks
Poor user-analyst communication and understanding
Poor or imprecise problem definition
Insufficient analysis when developing estimate
Poor estimating methodology or guidelines
Lack of coordination of functions (systems development, technical services, operations, data administration, etc.) during development
Changes in Information Systems Department personnel
Insufficient time for testing
Lack of setting and review of standard duration for use in estimating
Lack of historical data regarding past estimates and actuals
Pressure from managers, users, or others to increase/reduce the estimate
Inability to anticipate skills of project team members
Red tape
Users' lack of data processing understanding
Lack of project control comparing estimates and actual performance
Reduction of project scope or quality to stay within estimate, resulting in extra work later
Inability to tell where past estimates failed
Lack of careful examination of the estimate by management
Little participation in estimating by systems analysts and programmers
Performance reviews don't consider whether estimates were met
Lack of diligence by systems analysts and programmers
Removal of padding from the estimate by management

(Lederer & Prasad, 1995:129)

Table 3. Correlation with Inaccuracy Percentage

FACTOR	CORRELATION
Management Control	0.41
Methodology	0.24
Politics	0.23
User Communication	0.14

(Lederer & Prasad., 1995:132)

necessary to provide a product that meets performance, schedule, and cost criteria. The Air Force and DOD have taken the stance that it is a science. This is evident when DOD directives and guidelines concerning acquisition are reviewed. The military has always supported regulated procedures, since DOD is in the business of fighting wars and protecting American beliefs, ideals, and freedom. The use of regulations, and now directives, is visible at all levels and agencies throughout the military. In the Guidelines for Successful Acquisition and Management of Software-Intensive Systems (an Air Force publication), it states, that "software acquisitions fail because software management fails! Software management fails in three areas: administration, program measurement, and technical scrutiny" (DAF, 1996:Sec 1, 18). Other reasons listed for software program failures include (DAF, 1996:Sec 1, 18-32):

1) Software complexity
2) Inadequate estimates, including size and complexity estimates; cost/schedule estimates; optimistic estimates
3) Unstable requirements due to lack of user involvement; communication; intangibility; complexity; changing threat
4) Poor problem solving/decision-making; there are no silver bullets

From the study done by Lederer and Prasad, there are some similarities between causes for inaccurate software estimates; however, Lederer and Prasad further determined that in information systems development, the user is not the primary issue; management is the primary issue (Lederer and Prasad, 1995:132-133).

Experience Level. An issue not normally identified in most literature concerning inaccurate estimating with software cost models is the experience level of the user with the particular model. In an interview with Brad Donald who is in charge of the Research and Contracts Division of the Air Force Cost Analysis Agency (AFCAA), he stated that "most software cost estimates are done by junior grade officers with the least experience. There may be only a dozen experienced software cost estimators in the Air Force" (Donald, 1997). DeMarco also noted there was a lack of development of estimating expertise (DeMarco, 1982:9). Donald also pointed out that "it seems that individuals will always tend to use the first cost model they ever used for all projects" (Donald, 1997). Both of these statements violate the findings of previous AFIT studies which state that model users need to become familiar and experienced with specific models, and that no specific model works best with all applications. To develop credible and accurate estimates requires experience and understanding of a model and the realization that some models are better at projecting costs for certain applications than others.

The fact is that software development continues to overrun cost and schedule. This is further perpetuated because almost every Air Force program (aircraft, communications, command and control etc.) requires software. In a 1990 Pentagon software research study on 82 large military procurement programs, the researchers "...found that programs relying heavily on software ran 20 months behind schedule—three times longer than non-software-intensive programs" (DAF, 1996:Sec 1, 6).

Capability Maturity Model. To address software issues, the Air Force believes that "an award to a contractor with a mature, well-defined, standardized process can translate into substantially lower program risk and cost savings for the Government through reduced documentation, oversight, review, and auditing requirements" (DAF, 1996:Sec 7, 5). A mature process is best described using the Capability Maturity Model (CMM) developed by Carnegie Mellon University and the Software Engineering Institute, an organization dedicated to the advancement and support of the software engineering community. The CMM is "a description of the stages through which software organizations evolve as they define, implement, measure, control, and improve their software processes (Carnegie Mellon University (CMU), & Software Engineering

Institute" (SEI), 1995:353). The CMM is broken down into five separate levels and their associated characteristics as follows (CMU, et al., 1995:15-17, 33).

1) Initial: *few processes are defined, success dependent upon individual efforts*
Ad hoc process

2) Repeatable: *basic project management processes are established; track cost, schedule and functionality*
Requirements management
Software project planning
Software project tracking and oversight
Software subcontract management
Software quality assurance
Software configuration management

3) Defined: *the software process activities are documented, standardized, and integrated; projects use approved, tailored version of organization's standard software process*
Organization process focus
Organization process definition
Training program
Integrated software management
Software product engineering
Inter group coordination
Peer reviews

4) Managed: *detailed quality and process measures (metrics) are collected for quantitative assessment and control*
Quantitative process management
Software quality management

5) Optimizing: *continuous improvement through feedback; piloting innovative ideas and technologies*
Defect prevention
Technology change management

Within an organization, the CMM is useful in identifying areas for improvement. When outside an organization, the model aids assessment of an organization's capabilities and puts them in perspective with other organizations. The Air Force recommends use of this model, which appears to be in line with the overall Air Force philosophy and Total

Quality Management (TQM). The major theme of the revised version of DOD 5000.2 (now 5000.2-R), a major acquisition publication, is that of teamwork through use of Integrated Product Teams (IPTs), empowerment, Cost As an Independent Variable (CAIV), the use of commercial products, and the Best Practices Initiative. Examples of Best Practices include: replacement of government-unique management and manufacturing systems with common, facility-wide systems; realistic cost estimates; best value evaluation and award criteria; identifying management goals, requiring reporting, and offering incentives; and an open systems approach, emphasizing commercially supported practices, products, specifications, and standards (DOD, 1996:9).

DOD and Industry Comparison. Although Air Force, and; therefore, DOD, weapons system development (which includes software development) appears dismal and destined for cost and schedule overruns, it should be put into perspective with the rest of industry. "The DOD is bound to get lots of public scrutiny, and bound to make some mistakes. It implements over 15 million contracts each year (52,000 each day), and it spends around $300 billion a year" (Gansler, 1989:4). "In comparison with many other organizations, the DOD does a relatively good job of controlling cost overruns" (Gansler, 1989:171). For example, the chemical, drug, public utilities (water and energy), and large construction industry average higher overrun costs than the DOD average of 40% (Gansler, 1989:5). For example, the New Orleans Superdome had an overrun of approximately 225% while some energy process plants averaged about 180% (Gansler, 1989:5).

Industry Viewpoint. The Air Force and DOD appear to have the same goal of trying to facilitate improvement of the software development process by implementing TQM with better measures and the Best Practices Initiative. Ultimately, this should assist in software cost and schedule estimation accuracy. On the other hand, the software engineering industry seems to be headed in several directions. Several theories (many unsupported by empirical evidence) conceived to solve the software cost model estimate inaccuracy include: inadequate risk analysis, management control, lack of quality management, lack of historical data to calibrate the model, inaccurate sizing methods, and the model itself.

Risk Assessment. According to one author,

> The problem most software development methodologies experience is they do not address risk: they do not identify project risks and act on them. Without the knowledge of risk management concepts inherent in the software development process, the ability to identify, plan, assess, mitigate, report, and predict risks is almost impossible. (Karolak, 1996:10-11)

Since software development is an intellectual activity, it is difficult to communicate requirements and direction, integrate the software, locate defects, and debug the code (Karolak, 1996:10). Therefore, a method to identify the risk and determine its impact upon the project is necessary. Risk on any project can be divided into three groups: technical/engineering, requirements, and cost estimating. Cost estimating risk deals with the error in the estimate due to inadequate or lack of historical data, estimating methodology, and simple data entry errors in the cost model parameters. Requirements risk deals with the threat of budget cuts, changing the schedule to meet an enhanced threat, or a user change due to a lack of understanding of what they thought they needed. Technical risk is the inability to deliver at a specified time or due to poor coding. The newer the technology or more complex the system, the greater will be the technical risk.

The first step in risk management is to identify it and determine possible impacts on the specific cost elements. For example, if it's estimated that a software project will require 100,000 SLOC, then the next question that needs to be addressed is "what is the worst and best case scenario"? To properly address the risk, a probability distribution function (PDF) can be derived by answering this question for each cost element that makes up the cost of the software. The type of PDF chosen for each cost element (i.e.

coding, administration, travel, etc.) can be correlated with other cost elements. The more common PDFs used in cost estimating include: normal or Gaussian, triangular, beta, and lognormal. Also, it's recommended by statisticians to correlate the cost elements because this can increase the variance of the aggregate PDF for the system. Most likely, the PDFs and correlation values will be subjective, just as it is for other development efforts. Once all the PDFs are created and corresponding mathematical relationships between the elements determined, an output PDF can be derived by using Monte Carlo simulation. This will result in a more credible estimate with an infinite number of associated probabilities. The estimator and/or program manager is then left with choosing between a cost associated with their choice of probability (i.e. a cost of $2M and 60 percent confidence or $2.5M and 75 percent confidence). Software cost model developers could incorporate this into their models. Some cost model developers, like Galorath Associates who developed SEER-SEM, give the user the option of choosing the type of risk analysis. PERT and Monte Carlo simulation are two of the several choices available in their model. Since SEER-SEM is proprietary, the extent to which the risk analysis is incorporated is not known for sure. Unfortunately, risk analysis has not proven to be the sole answer to providing an accurate cost model.

Configuration Management. At the most recent DOD Cost Analysis Symposium in Williamsburg, VA, in February 1997, two topics generating a large amount of interest and discussion were applying risk analysis to cost estimates in general and inclusion of a management parameter in software cost models. In 1984, Edward Bersoff, a senior member of the IEEE, published an article recognizing the importance of configuration management. Bersoff classified identification, control, auditing, and status accounting as activities that constitute software configuration management (Bersoff, 1984:82). Identification includes the labeling of baseline components, which allows for careful monitoring. Control provides "...the administrative mechanism for precipitating, preparing, evaluating, and approving or disapproving all change proposals throughout the system life cycle" (Bersoff, 1984:82). Automated tools such as Program Support Libraries (PSL) support the control function by keeping a copy of each authorized version of software configuration items. Auditing provides the means for actual and baseline activities to be compared. Software metrics are a means of auditing a software project and may be defined as "...a measurable indication of some quantitative aspect of a system. For a typical software endeavor, the quantitative aspects for which we most require metrics include scope, cost, risk, and elapsed time" (DeMarco, 1982:49). A metric is useful if it is measurable, can't be influenced by personnel (independent), accountable, and precise" (DeMarco, 1982:50). Metrics can be divided into either a result or predictor metric. A result metric relates to the completed system for cost, manpower, performance and a predictor metric is one that has a strong correlation with a future outcome, such as complexity" (DeMarco, 1982:54). Status accounting is the administrative mechanism for the tracking of software identification components, control items, and auditing results. Software cost model developers have been increasingly including input parameters for management within the models in some form or another. Some are direct inputs, like management ability, while others are indirect through some other input, such as team capability. Overall, the greater the awareness management has of the development process and the action they take to remedy the situation should equate to a higher quality product that is produced in a shortened period of time at less cost.

Total Quality Management. Another aspect of management is quality management. In the Air Force, it's recognized as Total Quality Management (TQM). In some instances, TQM has been deemed the panacea for any problem, while in other instances, it has been treated as the scapegoat for a failure. Realistically, the question arises as to the validity of TQM. The term "quality" itself means different things to different people and entities. For a consumer, quality may take the form of a product that meets the consumer's expectations. For an organization, it may be the most cost effective product. In the Air Force, a quality product is one that meets cost, performance, and schedule criteria. According to Philip Crosby, "quality is free. It's not a gift, but it is free. What costs money are the unquality things—all the actions that involve not doing jobs right the first time" (Crosby, 1979:1). Rework, scrap, warranty service, and inspection are all results of nonconformance to quality. These types of services tend to be necessary due to poor design. It has been estimated that "the design phase of a project is responsible for 85 percent of life cycle cost commitments" (Brabson, 1982:46). Crosby's idea is that with proper planning, integration, and employee involvement to identify issues and risky situations, we can avoid a large amount of the cost (Crosby, 1979). But, the organization must be willing to forego this up-front cost of time and/or money to achieve the savings downstream. According to Crosby, a manager should display certain characteristics. Some of these include: integrity, compassion, listening, helping, cooperating, learning, leading, and following" (Crosby, 1979:146). A manager must be able to recognize the resources he has and allow them to do what they do best.

> The super designer or super programmer can make a mediocre crew do great things, if given the chance. Such a person can: teach others how to use the available software tools; provide on-the-job training while supervising the actual work; ensure the software design is really good and instruct the programmers in how it works; inspire others with the example of high achievement and an enthusiastic approach. It is a lucky firm that has one such person for every ten other people. It is a wise firm that knows his value. (Softkey, 1983:7)

Metrics can enhance quality because they help managers and employees to determine how they are doing. According to DeMarco, the defects metric (an excellent software quality metric) is the only metric that should actually be collected on a continual basis (DeMarco, 1995:15). Other metrics should only be collected on a short term basis. DeMarco also pointed out that many metrics have not yet been empirically confirmed, including Halstead's proposed metrics in his book Elements of Software Science, written in 1977 and the very popular McCabe's Cyclomatic Complexity V(G) metric (DeMarco, 1995:30-32). Overall, management can be overloaded with the number of metrics available to measure software development. The only metrics worth using are those that measure for benefit and discovery (DeMarco, 1995). According to Goel, the software reliability metric is the best method of quantifying software quality (Goel, 1985). Intuitively, TQM appears to be a critical organization philosophy. Supporters of TQM would insist that those organizations (such as software developers) that practice TQM, will incur lower costs, improve their products, enhance market share, and improve employee morale. One fact is known for sure; quality management works for the Japanese, who are now the leaders in many industries that were once led by U.S. companies.

Calibration. Previous studies have proven repeatedly that calibration will improve software estimating accuracy. Unfortunately, calibration requires standardized historical data. For software programs in the Air Force, data is plentiful; however, once the data is stratified, the analyst is left with very little to work with and the data is full of holes. The software industry is experiencing the same problems. "Except in the most successful projects, everyone scurries off at the end without even taking note of the actual total cost. Estimates for the next project are made as though the last project never happened, and no one benefits from past mistakes" (DeMarco, 1982:5-6). This lack of data to calibrate models not only affects the estimate, but doesn't allow for learning from past mistakes. "The only unforgivable failure is the failure to learn from past failure" (DeMarco, 1982:6). Without appropriate milestone data, defect and reliability rates, productivity rates, and other indicators of performance, an analyst can't provide management with benchmarks for future performance.

Estimating Size. Early in a software program development, many cost models use one of three indicators (SLOC, FPs, or Object Points) of program size to estimate cost. Unfortunately, coding does not normally begin until the Detailed Design phase, which is the point when the programmers can actually begin to give more accurate size estimates. The software industry is divided on which is the best method to indicate size. SLOC is the most widely used method to indicate size, but their is growing interest in FPs and more recently, Object Points (Boehm, Presentation, 1997). According to various authors, there are seven reasons that it's difficult to project cost estimates from SLOC (or program size) early in a program:

1. Size is affected by language, application area, software complexity, design methodology, programmer style and capability. (Lokan, 1996:65)
2. There is no obvious relationship between SLOC and the end product. (Dolkas, Evans, and Piazza, 1983:143)
3. Size is not a consistent indicator. As language changes, SLOC changes; there is no standard to help normalize between programs. (DeMarco, 1982:29)
4. There is a lack of support by programmers as to the significance of SLOC and cost estimating. (Dolkas, et al, 1983:143)
5. "...There are many ways a set of specifications can be coded to achieve the same basic result, even when the input and output are fixed. (Dolkas, et al, 1983:143)
6. There is a general lack of understanding by the user and developer of what actually must be done. (Dolkas, et al, 1983:143)
7. Over half of the activities involved in software development are not affected by the language, and therefore, the size of the program. (Jones, 1986:7)

As a project gets closer to completion, especially during and after coding, SLOC estimates become more accurate, which enhances the accuracy of the estimate. Early in the program, SLOC is determined from historical data and expert opinion, but, if there is any substance to reason four listed above, the expert programmers may not be putting much thought into their SLOC estimate.

An alternative method of estimating size is to use FPs, which are based on the number of inputs, outputs, files, and queries the software must handle. The advantage of this method is that it does not require a determination of the estimated SLOC. FPs rely on understanding what the user needs the program to do. The limitations of FPs follow (Boehm, Presentation, 1997):

1. Ability to estimate real time and highly complex software.
2. Like SLOC, inputs necessary for FPs not always available early in a program.
3. FPs are difficult to understand.

To alleviate the above issues, the International Function Point Users Group (IFPUG) was formed and is dedicated to standardizing and promoting the use of FPs.

Another alternative to SLOC is Object Points, which is a variant of FPs. Object Points is gaining popularity because it promotes modularity, is easier to understand than FPs, is good to use with CASE Tool development, and it provides a means to measure effort directly (Ferens, 1997). On the downside, it is still in the research stage, so it hasn't been proven yet, but nor have FPs or SLOC been proven highly successful. Object Points seem like a promising estimating parameter, in fact, Dr. Boehm is emphasizing Object Points in his research and has included it for use in the early design mode of the COCOMO II model (Boehm, Interview, 1997).

Design Methods. Since size is difficult to estimate early in a program's development, the choice of design methodology may be a critical factor to enhancing size estimates, and therefore, improving software cost estimates. The Air Force is highly interested in Object Oriented Design because weapons system software is highly complex (DAF, 1996:i-iii). A weapon's system complexity is due to size, real time nature, and algorithmic makeup (Ferens, 1997). Object Oriented Design should enhance the programmer's ability to estimate object points and SLOC because of its modular methodology. However, even when the developing team uses Object Oriented Design, software cost models have still not been successful in creating an accurate estimate consistently in DOD. Ada was developed as an object-oriented language (even before the term was known) and designed to support reuse and COTS (DAF, 1996:iii). According to Lloyd Mosemann, Deputy Assistant Secretary of the Air Force for Communications, Computers, and Support Systems, Ada is the language of choice for weapons systems (DAF, 1996:iii). Nevertheless, even with the use of Ada and Object Oriented Design, software cost models still have not shown improved estimating ability.

Due to development issues such as consistency and product quality, a new modeling language has been developed. The University of Southern California, Center for Software Engineering, directed by Dr. Boehm, is investigating this new modeling language. It's identified as Unified Modeling Language (UML) and is an alternative to using FPs, SLOC, and object points in determining size estimates (Boehm, Interview, 1997). UML is a "...collection of 'best engineering practices' that have proven successful in the modeling of large and complex systems. In the same way that a blueprint helps a team collaborate successfully on constructing a building, the UML helps a team visualize an application's architecture throughout the development lifecycle" (Rational Software Corporation (RSC), 1997). This new graphical language is based on the best and most useful characteristics of modeling languages of leading object oriented methods (RSC, 1997). It addresses factors such as concurrent development and distributed systems and it focuses on a standard modeling language versus a standard process (RSC, 1997).

The value of UML is that "it removes the unnecessary differences in notation and terminology that obscure the underlying similarities of most of these approaches," which, has been noted by some authors of software engineering as one of the issues clouding software development estimates (RSC, 1997). UML is being submitted to the Object Management Group (OMG) for adoption.

OMG was established as a non-profit corporation in 1989 to "promote the theory and practice of object technology for the development of distributed computing systems" (OMG, 1997). OMG has a "...commitment to developing technically excellent, commercially viable and vendor independent specifications for the software industry, the consortium now includes over 700 members" (OMG, 1997). This concerted effort by the software industry may help to solve some of the issues surrounding standardization of software development. However, there is doubt that this will occur, because the concerted efforts of IFPUG have been unsuccessful in establishing FPs as a standard in the software industry.

The Software Cost Models. The last primary cause of cost estimating inaccuracies, as viewed by the software industry, is the software cost models themselves. Each model developer has his trademark. Dr. Jensen, the developer of SAGE, emphasizes the importance of management. On the other hand, Dr. Boehm downplays it somewhat because he doesn't want to reward poor management with a higher estimate by having a direct input into the COCOMO II model (Boehm, Interview, 1997). The SEER-SEM developers have taken the approach to include over 30 input parameters, including the ability to run Monte Carlo simulation to compensate for risk (Galorath Associates, Inc., 1996). On a larger scale, the CHECKPOINT model, which was developed by Capers Jones of Software Productivity Research, Inc. (SPR), includes over 100 input parameters (SPR, 1993). However, even with these differences, there are some similarities between the software cost estimating model inputs. Some of these parameters include: programmer and analyst capability, multisite development, automated tools, programmer and analyst experience, language used, application type, and system volatility.

When developing a functional relationship between the independent (model input parameters) and dependent (effort, schedule, or size) variables, there are several methods that may be used; these methods include: analogy, top-down, bottoms-up, expert opinion, and regression. These methods may be used individually or in any combination. Many of the models use regression, also known as a cost estimating relationship (CER), for developing this functional relationship.

When developing the CER, there are two approaches that may be used. The first is to take a logical approach and only include dependent variables that logically have a relationship to the dependent variable, for example, use software application as one of the independent variables to determine effort required to complete a software program. The second approach is to use any variable that improves the explanatory power of the model, as represented by the coefficient of multiple determination (R^2). An example of this may be the use of platform and language to help determine the effort required to complete a software program. Some may feel that platform encompasses (highly correlated with) the effects of language, and to include language as a parameter only serves to improve the explanatory power of the model. In addition, it's extremely important to understand that high correlation between dependent and independent variables does not necessarily imply a reason to include a dependent variable in a regression equation; this may be the argument for not using SLOC to determine effort and development time. An example of this would be to use the rise in the use of cellular phones to project the number of cancer patients.

It appears that some software cost model developers take the (yet illogical?) approach by using a high number of input parameters. Since many models are proprietary, assessment of the internal equations is not possible, but, a high number of input parameters in a cost model may indicate the illogical approach to achieve a higher

R^2. The reason for including all of these parameters though may best be summed up by Dolkas, Evans, and Piazza:

> Many technical solutions have been heralded as panaceas for development. Structured design and development, top-down testing, automated development aids, software quality assurance, and many other tools and techniques all have assumed important roles in the development process; none by itself, however, has solved the fundamental software problems. We still do not know how to develop quality software consistently within cost and schedule. (Dolkas, et al, 1983:1)

Unfortunately, it's still not possible to determine "why software cost estimating models are inaccurate?" The literature has identified several ways to improve the software process, but no one has yet proven that their ideas are successful. The main themes in the literature were to: document, collect useful and standardized data, plan to plan, follow a plan, use configuration management, use software assurance and automated tools, apply good management techniques, hire good personnel, and understand that change is inevitable.

COCOMO II

Overview. In this section, a general comparison of the differences between COCOMO 1981 and COCOMO II will be highlighted. This will be followed by a discussion of the weaknesses of COCOMO II. Lastly, the model equations will be presented and analyzed.

Comparison of COCOMO 1981 TO COCOMO II. COCOMO II is very similar to COCOMO 1981. The theory surrounding the models has been modified to keep pace with current and future trends, but the basics have not changed. As before, there are three estimation stages of the model. One difference between the two models is that the original COCOMO used SLOC to determine size, whereas COCOMO II has incorporated the use of Object Points, FPs, and SLOC. Stage one (prototyping) uses

Object Points to calculate size because it does not rely on the use of SLOC and "from a usage standpoint, the average time to produce an object point estimate was about 47 percent of the corresponding average time for function point estimates" (Boehm, Clark, Horowitz, Westland, Madachy, & Selby, 1996:5). Stage Two (Early Design) and Stage Three (Post-Architecture) of the model allow use of FPs or SLOC. It's important to note that FP inputs are converted to SLOC using Capers Jones FP/SLOC conversion chart (Boehm, Presentation, 1997). Modifiers for reuse and software breakage have also been included. Software breakage is the percentage of software thrown away due to requirements volatility (USC, Reference Manual, 1997:2).

The most notable change to COCOMO is that its been adapted to the Microsoft Windows environment and is extremely user friendly. There have also been changes to the input parameters (effort multipliers). Intermediate COCOMO 1981 and Post-Architecture COCOMO II are compared in Table 4 on the following page. Several of the effort multipliers were combined due to high correlation, and others were added because it was determined necessary to incorporate them within the new model (Boehm, Presentation, 1997). Virtual machine volatility was replaced with the platform volatility multiplier, whereas, the turnaround time multiplier was dropped. According to Dr. Boehm, turnaround time was no longer necessary because of the interactive systems now available (Boehm, Interview, 1997). Virtual machine experience was replaced by platform experience. Language experience was replaced by language and tool experience. Modern practices were replaced by both platform experience and tool experience (Boehm, et al, 1996:14). Documentation, reusability requirements, and multiple site development were all added due to their importance in today's software development. Most of the effort multiplier values went unchanged or only had minor changes, but the Size exponent in the equation has been enhanced.

Scaling Factors. The original COCOMO model had a fixed size exponent for each mode (1.05 for organic, 1.12 for semi-detached, and 1.20 for embedded), COCOMO II now includes scaling factors which determine the actual exponent value and can vary from 1.01 for all extra high ratings to 1.26 for all very low ratings. The scaling

Table 4. COCOMO Input Parameter Comparison

EFFORT MULTIPLIER	COCOMO 1981	COCOMO II
Required Reliability	RELY	RELY
Data Base Size	DATA	DATA
Product Complexity	CPLX	CPLX
Memory Constraints	STOR	STOR
Timing Constraints	TIME	TIME
Virtual Machine Volatility	VIRT	
Turnaround Time	TURN	
Analyst Capability	ACAP	ACAP
Programmer Capability	PCAP	PCAP
Analyst Experience	AEXP	AEXP
Virtual Machine Experience	VEXP	
Language Experience	LEXP	
Modern Develop Practices	MODP	
Use of Modern Tools	TOOL	TOOL
Schedule Effects	SCED	SCED
Documentation		DOCU
Required Reuse		RUSE
Platform Volatility		PVOL
Platform Experience		PEXP
Language/Tool Experience		LTEX
Personnel Continuity		PCON
Multiple Site Development		SITE

factors were based on those from the Ada COCOMO model (USC, COCOMO II Model Definition, 1997:16). "The selection of the scale drivers is based on the rationale that they are a significant source of exponential variation on a project's effort or productivity variation" (USC, COCOMO II Model Definition, 1997:16). These added scaling factors and effort multipliers increase the sensitivity of the model.

The scaling factors are (USC, COCOMO II Model Definition, 1997:16-20):

1. Precedentedness - PREC; identifies the newness of the project
2. Development Flexibility - FLEX; degree of requirements, schedule, interface, etc. flexibility
3. Risk Resolution - RESL; degree of risk present
4. Team Cohesion - TEAM; project turbulence and entropy of the project team
5. Process Maturity - PMAT; uses CMM questionnaire to determine weighted average.

Annual Update to COCOMO II. In a presentation to the AFCAA, Dr. Boehm presented several trends that will create difficulties in software cost estimating. They include (Boehm, Presentation, 1997):

1. Graphic User Interface builders, Commercial-Off-The-Shelf (COTS), Fourth General Languages (4GL), reuse, and breakage;
2. Distributed interactive applications; e.g. middleware effects (cut and paste from the Internet and other programs;
3. New process models such as evolutionary, incremental, and spiral; may induce phase overlap and new labor distributions (versus Rayleigh curve).

To overcome some of these trends, COCOMO II will be updated on an annual basis. USC is continuously gathering new data to recalibrate the model. The current version of COCOMO II is calibrated to 83 data points, some of which are from the SMC SWDB. This could improve the accuracy of the estimates in the uncalibrated mode for this study.

Model Weaknesses. Several model weaknesses were identified during a personal interview with Dr. Boehm. Several of these will be emphasized in future research and updates.

The current weaknesses of COCOMO II are as follows (Boehm, Interview, 1997):

1. The model does not take into account the different types of software development process models (Waterfall, Incremental, Spiral, Evolutionary), it is still based on the Waterfall model; there are plans are to reanalyze;
2. Currently, there is no on-line calibration; a new version with on-line calibration is due out in the summer of 1997;
3. There are no risk related outputs or inputs (except for PERT style FP input) that could take advantage of a Monte Carlo simulation; there are plans to investigate;
4. There are no defect estimations; this is currently under research;
5. The security parameter is not included; however, user can add the security parameter in themselves under one of two user defined parameters;
6. There is no language input at this time; language adaptation is currently being developed for addition;
7. Estimates are based on SLOC, however, only 30 to 40 percent of the schedule and cost may be attributable to SLOC. Typically, SLOC has demonstrated a high correlation with effort, but with the new technologies and techniques, effort estimation will require some other method;
8. No maintenance/support estimate is calculated; they will add this at a later date;
9. Reports cannot be printed directly from the model; this may be added later.

Although there are several weaknesses with the model, Dr. Boehm reports that for the 83 data points used to calibrate COCOMO II, the model's estimates were within 30 percent, 66 percent of the time (Boehm, Presentation, 1997). This is not a poor level of accuracy since the database contains a diverse set of applications. For this study, the model will be calibrated to more specific applications in some instances, as explained in Chapter III.

Model Equations. There are three basic equations in the COCOMO II model. One estimates size, another estimates effort, and the last estimates schedule. COCOMO II is an algorithmic model, which simply means that the estimate is derived from a functional relationship with one or more variables. It's also what is considered a composite model because it uses a combination of linear and multiplicative relationships to derive the estimate. The first equation is:

$$\text{Size} = \text{KNSLOC} \quad (1)$$

$$B = 1.01 + 0.01\ \Sigma^{5}_{j-1}\ SF_j \quad (2)$$

where KNSLOC equals the size of the component expressed in thousands of new SLOC. The second equation is the Size parameter exponent. A nominal value of the B component for COCOMO II is 1.16, whereas, for COCOMO 1981, the exponent was fixed at 1.12 for the semi-detached mode. The next equation calculates Person Months (PM) from the Effort Multipliers (EM) and the Scaling Factors (SF).

$$PM = \Pi^{17}_{j-1}(EM_i) * A * [(1+BRAK/100)*Size]^{B} \quad (3)$$

Where BRAK equals the percent of code thrown away due to requirements changes. The constant 'A' and 'B' are normally set at specific values, but those numbers will be calibrated to the specific data sets. Once the effort has been calculated, the development time can be determined. The schedule (TDEV-time to develop in months) can be calculated from the Person Months.

$$TDEV = [3.0 * (PM)^{(0.33 + 0.2 * (B-1.01))}] * SCED\%/100 \quad (4)$$

SCED% is the compression of or expansion of the schedule from what is considered nominal (USC, COCOMO II Model Definition Manual:14). Due to the exponential nature of B, we can see that the scale factors will have a significant impact on an estimate. Experts support the significance of the user's understanding of how a cost model functions, because it can give the user insight to sensitive variables and specific relationships. For this research, the scaling factors will not have an impact because the SMC SWDB does not include the necessary data to determine the scaling factors.

Summary

This chapter has provided a cursory background of the software engineering and cost estimating field. The ideas presented are from some of the most prominent experts in the field. Intuitively, many of the theories seem relevant and credible, but many have not been empirically proven. This background information has provided the necessary understanding to make an objective analysis of the COCOMO II model. The problem of cost and schedule overruns in software development has been unchanged for the past 20 years. It's reasonable that "software cost-estimation techniques are important because they provide an essential part of the foundation for good software management. Without a reasonably accurate cost-estimation capability, software projects experience the following problems: proposed budget and schedule is unrealistic, no means of making realistic tradeoff analysis during design phase, and no basis for determining individual phase duration and effort" (Boehm, 1981:30).

III. Methodology

Overview

The objective of this chapter is to explain the elements of the actual research. The COCOMO II software cost and schedule model will be used to estimate the effort of projects stored in the SMC SWDB. These estimates will then be compared to actual values. The results will be analyzed to determine overall effectiveness of the model. To aid in understanding the process, a discussion of the SMC SWDB and the procedures to stratify the data will be presented, followed by a step-by-step description of the proposed calibration of the model. Lastly, the chapter will conclude with a discussion of the methods used to validate and analyze the results of the model runs.

Procedures and Data Analysis

SMC SWDB. The SMC SWDB was first established and assembled by Stukes of MCR Incorporated in 1989. It was originally compiled to provide a means to calibrate the following software cost models: PRICE-S, SASET, and SEER-SEM (Apgar, Galorath, Maness, and Stukes, 1991). The mission now has been expanded to include other models of interest to the Air Force. The database is based in FoxPro (a Microsoft database program) and allows the user to accomplish multiple queries on the database. The SMC SWDB, Version 2.1, which will be used in this analysis, contains 2,637 records in various military and commercial applications, such as avionics, military ground, space, unmanned flight, and management information systems (MIS). Some of the records are reported at the CSCI (Computer Software Configuration Items), each distinguished by 276 data parameters which match commercial models and cost structures. Data sources include, but are not limited to SMC, European Space Agency, and National Aeronautical Space Agency programs managed by Air Force Material Command, Goddard Space

Center, Jet Propulsion Laboratory, General Dynamics, some major Aerospace Companies, and some non-aerospace companies such as American Telephone and Telegraph.

Once data are received for the database from contractors, suppliers, and Air Force and other DOD agencies, MCR maps and normalizes the data. Adjustments to the data are made based on inflation, economies of scale, technology, design year, new versus upgrade, and whether it's an incomplete system (Stukes & Patterson, 1996). Except for effort, details of these adjustments are not given. Fortunately, the data required for this research should not be affected by any adjustments except for effort, which will be discussed later in this chapter. Each record may include project description (but not the company name), size metrics (SLOC and some Function Points), schedule metrics, effort metrics (by phase or by labor category), and complexity metrics (personnel, tools, environment, and standards—based on DOD-STD-2167 and MIL-STD-498) (Stukes et al, 1996). There is a composite of 50 million SLOC which includes new SLOC and equivalent SLOC. The equivalent SLOC was normalized from percent of reused code (Stukes et al, 1996).

SMC SWDB Query Setup. For this research effort, it will be necessary to stratify the data to establish consistency. The queries are consistent with a 1996 thesis by Southwell, and will be as follows (Southwell,1996:28):

1. Software Level = CSCI
2. Software Functions = All
3. Programming Language = All
4. Effective Size Range = 2,000 to 300,000 SLOC (not to include records with this field empty); Dr. Boehm states that estimating with the model for anything less then 2,000 SLOC is ineffective (Boehm, Presentation, 1997)

5. Total Size Range = 1 to 9,999,999,999 (not to include records with this field empty)
6. Effort Range = 1 to 9,999,999,999 (not to include records with this field empty)
7. Years of Maintenance = 0 to 9,999,999,999 (to include records with this field empty)

The queries will also be limited to the following categories of records shown in Table 5:

Table 5. SMC SWDB Queries

Query Title	Operating Environment	Applications
Military Ground - C^2	Military Ground	Command & Control
Military Ground - SP	Military Ground	Signal Processing
Unmanned Space	Unmanned Space	All
Ground in Support of Space	Ground in Support of Space	All
Military Mobile	Military Mobile	All
Missile	Missile	All
Mil-Spec Avionics	Mil-Spec Avionics	All

(Southwell,1996:29)

There are two objectives when running the query. The first objective is to obtain a minimum of 12 data points if possible, which will improve the calibration of the coefficient of the model, thereby, enhancing the model's ability to estimate more accurately. According to Clark, the optimal number of data points to actually calibrate the effort coefficient is 10 or more (Clark, 1997). This equates to 12 data points per query title using the resampling method (described later in this chapter) to validate the model. The second objective is to arbitrarily obtain a minimum of four sets of data. Since the resampling method is used, a data set may have as few as four data points; nevertheless, for purposes of calibrating the model, a minimum of 12 data points will be strived for. Based on previous experience with the database, it's anticipated that to generate a minimum of four data sets, it may be necessary to lower the number of data points to eight (Ferens, 1997).

Performing the Query. To help simplify the query process, follow the steps as listed below:

1. Set up the query in accordance with Table 5 and as outlined on pages 37-38.
2. Run the query on the SMC SWDB.
3. Determine the number of projects (identified as keyfields in the database) per query, discarding all European projects since they are not consistent in their development with U.S. projects.
4. If there are an unusually large number of projects (> 25), then try to limit the projects to one homogeneous application, such as Signal Processing, so that there are still 12 projects or more.
5. If the query has less than 12 projects, then set those queries aside at this point.
6. If at least four queries out of the seven queries listed in Table 5 generated 12 projects, then determine whether each project has listed Total Normalized Effort. If so, then go to Further Analysis below, if not, then continue.
7. This step applies only to the Military Ground queries, otherwise go to step 8. If at least four queries out of the seven queries listed in Table 5 did not generate 12 projects, then change the 'Application' type (e.g. change Signal Processing to All) of those queries with < 12 projects, in order to generate 12 projects per query.
8. If no less than four queries with 12 projects are generated, then determine that each project has listed Total Normalized Effort. If so, then go to Further Analysis below, if not, then continue.
9. Use all queries that generated at least 12 projects, and also use the queries with the greatest number of the projects, to at least have four queries. If a query has < 4 projects, it is not usable, based on the resampling method.

Further Analysis. Once the queries have been completed as described, further analysis of the projects will be necessary to discern any abnormalities. An example of this would be a complex program that shows a programmer capability rating of low. Intuitively, it doesn't make sense that an organization would put inexperienced or ineffective programmers on a complex project, nor would the government contract out a job to someone if they felt the programming capability was low for a complex project (Clark, 1997). If there appear to be abnormalities for a specific parameter, then that parameter will be left as a default (nominal) parameter for all data points within that application (Clark, 1997). Likewise, if not all parameters within a project contain an entry (e.g. analyst capability), then that parameter for all the data points within the application will also be left as default (nominal). This applies to the calibrated estimate only.

SMC SWDB Weaknesses Identified. Before the actual data stratification and analysis, several apparent weaknesses of the SMC SWDB must be identified.

1. This version of the database was not set up using the COCOMO II model parameter descriptions for categories. Therefore, it will be necessary to compare parameter entry descriptions in the SMC SWDB to the actual descriptions in the COCOMO II Model Definition Manual.
2. Much of the data are contractor supplied data; therefore, the accuracy of it is questionable. To further complicate the data, the contractor(s) executing the projects are not identified within the data base to keep their anonymity. According to Brad Clark, a Ph.D. student from USC working on the COCOMO II project, he found several abnormalities within the SMC SWDB (Clark, 1997). This can have an effect on the outcome of the analysis, such as lower accuracy.

3. Although MCR has gone to great lengths to request data in a standardized form, it's assumed that there will be some inconsistencies between organizational definitions and categorizations of data. Unless this is detected during data analysis, the data will be accepted as is.
4. The age of the data is unknown. Since some of the data in the database has been taken from other databases (e.g. Space Systems Cost Analysis Group) and other uncontrollable entities (e.g. contractors), the age of the data is truly unknown.
5. Due to the large number of records and fields in the database, there is an increased chance of data entry error.

Due to the first weakness listed, it may also be necessary to normalize the software phases and effort to match that of the COCOMO II model. This is easily done by adjusting the Total Normalized Effort from the SMC SWDB by the scaling factor listed in Table 6 as follows:

Table 6. SMC SWDB Software Phase Normalization

PHASE	% OF NORMALIZED EFFORT
Software Requirements	5.5
Preliminary Design	11.4
Detailed Design	19.1
Code and Unit Test	29.8
CSC Testing and Integration	35.6
CSCI Testing	4.1
Systems Test and Integration	7.2
OT&E	4.8

(Stukes, 1995:F-2)

The COCOMO II model bases its estimates on all phases beginning with Software Requirements through CSCI Testing. The Total Normalized Effort within the database, includes all phases from Preliminary Design through CSCI Testing, and is based on 152 hours per person month. Therefore, it will be necessary to add 5.5 percent of the Total

Normalized Effort for Software Requirements. Until actual data manipulation begins, the list of weaknesses will be assumed complete. If upon manipulation other errors or weaknesses are detected, they will be identified in Chapter IV.

Model Calibration. COCOMO II will be calibrated using the steps outlined in Chapter 29, pages 524 thru 530, of Dr. Boehm's book, Software Engineering Economics. Since the number of data points will be less than 100, only the *A* coefficient (multiplicative calibration variable) will be calibrated. There will be no attempt at calibrating the Effort Multipliers (EM) since there will not be enough data points to justify it. The *B* exponent (captures relative economies/diseconomies of scale) will be set equal to 1.153 because the database will not contain sufficient information to generate entries for the scaling factors (Ferens, 1997). Therefore, the scaling factors will be left nominal and the B exponent is calculated as follows:

$$B = 1.01 + 0.01 \Sigma^{5}_{j=1} SF_j \qquad (5)$$

The nominal values for all five scaling factors are different and are subject to change with each new COCOMO II version. The sum of the five scaling factors is 14.3.

$$B = 1.01 + 0.01 * 14.3$$

$$B = 1.153$$

As described by Dr. Boehm in his book Software Engineering Economics, the steps to calibrate the coefficient *A* of the Effort Equation is as follows (Boehm, 1981:525-526):

Step 1.

$$PM = A * SIZE^{1.153} * \Pi^{17}_{i=1} EAF_i \qquad (6)$$

Step 2.

$$PM_1 = A * SIZE^{1.153} * \Pi\, EAF_1$$
$$\Downarrow$$
$$PM_n = A * SIZE^{1.153} * \Pi\, EAF_n \qquad \text{where } n = \text{\# of projects}$$

This step minimizes the sum of the squares (S) of the residual errors to:

$$S = \Sigma^n_{j-1} [A (SIZE_i)^{1.153} * \Pi^n_1 EAF_i - PM_i]^2 \qquad (7)$$

Equation 7 may be simplified to:

$$S = \Sigma^n_{i=1} [A * Q_i - PM_i]^2 \qquad (8)$$

$$\text{where } Q_i = (SIZE_i)^{1.153} * EM \qquad (8a)$$

$$\text{and } EM = \Pi^n_1 EAF_i$$

Step 3.

Determine the optimal coefficient A by setting the derivative dS/dA equal to zero

$$0 = (dS/dA) = 2\ \Sigma^n_{i=1} [A * Q_i - PM_i]^2 * Q_i$$

$$0 = \Sigma^n_{i=1} [A * Q_i^{\ 2} - PM_i]* Q_i \qquad (9)$$

Step 4.

Solve for A using:

$$A = (\Sigma^n_{i=1}\ PM_i\ Q_i) / (\Sigma^n_{i=1}\ Q_i^{\ 2}) \qquad (10)$$

The data for calibrating the *A* coefficient will be recorded in the following format shown in Table 7:

Table 7. Calibrating the Coefficient *A*

Project	$SIZE_i$	EM_i	PM_i	Q_i	$PM_i * Q_i$	$Q_i^{\ 2}$

Calibrating the model to the specific applications should improve estimating accuracy; however, improvement may be limited since the model was originally calibrated using 83 data points, some of which were from the SMC SWDB.

<u>Resampling Method</u>. Once the data sets are established, the calibration and validation can be performed using the resampling methodology.

The steps in this process are listed below:

1. Randomly divide each data set into two subsets by using Microsoft Excel®, Version 7.0 random number generator.
    - The calibrated subset will contain 80 percent (or more, all fractions will be rounded up for the calibration data set) of the available data points.
    - The remaining data points will be entered into the validation subset.
2. The validation subset will be used to verify any improvement in accuracy between the uncalibrated and calibrated model. The model's ability to estimate accurately is expected to show some improvement with calibration.
3. Run the default (uncalibrated) model against each validated subset and record the results.
4. Calibrate COCOMO II model as described above using only the calibrated subsets from each query. This model will then be used to estimate effort for the validated subset of each application.
5. Repeat the previous steps, five times; each time selecting a new set of data points (a subset of the calibrated data set) using Microsoft Excel®, Version 7.0 random number generator, to calibrate the model.
6. This method, known as resampling, should produce more robust results to analyze, especially when using smaller data sets (Clark, 1997). Once the results are generated and recorded, the next step will be to analyze the results.

Analysis Methodology. The first step will be to apply Conte's criteria to determine the accuracy of the calibrated and uncalibrated model. This will be achieved using the following equations.

Conte's Criteria. *First, calculate the Magnitude of Relative Error (degree of estimating error in an individual estimate) for each data point. This step is a precedent to the next step and is also used to calculate PRED(ℓ). Satisfactory results are indicated by a value of 25 percent or less (Conte et al, 1986:172-175).*

$$MRE = |(\text{Estimate} - \text{Actual})/\text{Actual}| \qquad (11)$$

Next, calculate the Mean Magnitude of Relative Error (average degree of estimating error in a data set) for each data set. According to Conte, the MMRE should also have a value of 25 percent or less (Conte et al, 1986:172-175).

$$MMRE = (\Sigma\ MRE)/n \qquad (12)$$

where n = total number of estimates

Now, calculate the Root Mean Square (model's ability to accurately forecast the individual actual effort) for each data set. This step is a precedent to the next step only. Again, satisfactory results are indicated by a value of 25 percent or less (Conte et al, 1986:172-175).

$$RMS = [1/n * \Sigma\ (\text{Estimate} - \text{Actual})^2]^{0.5} \qquad (13)$$

Lastly, calculate the Relative Root Mean Square (model's ability to accurately forecast the average actual effort) for each data set. According to Conte, the RRMS should have a value of 25 percent or less (Conte et al, 1986:172-175).

$$RRMS = RMS / [\ \Sigma\ (\text{Actual})/n] \qquad (14)$$

A model should also be within 25 percent accuracy, 75 percent of the time (Conte et al, 1986:172-175). To find this accuracy rate PRED(ℓ), divide the total number of points within a data set that have an MRE = 0.25 or less (represented by k) by the total number of data points within the data set (represented by n). The equation then is:

$$PRED(\ell) = k/n \qquad (15)$$

where ℓ equals 0.25 (Conte et al, 1986:173).

Wilcoxon Signed-Rank Test. The next step will be to test the estimates for bias. The Wilcoxon signed-rank test is a simple, nonparametric test that determines level of bias. A nonparametric test may be thought of as a distribution-free test; i.e. no assumptions about the distribution are made (Conover, 1980:92). The best results that can be achieved by the model estimates is to show no difference between the number of estimates that over estimated versus those that under estimated. The Wilcoxon signed-rank test is accomplished using the following steps (Mendenhall, Wackerly, and Scheaffer, 1990: 680):

1. Divide each validated subset into two groups based on whether the estimated effort was greater (T+) or less (T-) than the actual effort.
2. Sum the absolute value of the differences for the T+ and T- groups. The closer the sums of these values for each group are to each other, the lower the bias.
3. Any significant difference indicates a bias to over or under estimate.

Regression Analysis. The results will also be analyzed using regression analysis. The method used will be similar to that reported by Kemerer in his article "An Empirical Validation of Software Cost Estimation Models" (Kemerer,1987). Each default estimate (independent variable) within a data set will be regressed against the actual effort (dependent variable) using Microsoft Excel® Version 7, and Statistical Analysis Software® (SAS). For the regression analysis, the data sets will be kept whole, and will not be separated into validated and calibrated subsets. If the resultant regression equation is not linear, then SAS® will be used to determine the best fit transformation. This can be accomplished by using the following statement when programming SAS®:

Model PMACT = PMDEF DEFSQR SQRDEF RECDEF/SELECTION=RSQADJ;	
where:	PMACT = actual effort for the project;
	PMDEF = estimated effort, default mode;
	DEFSQR = squared value of PMDEF;
	SQRDEF = square root value of PMDEF;
	RECDEF = reciprocal value of PMDEF;
	RSQADJ = adjusted coefficient of (multiple) determination;

Figure 1. Regression Variables

The benefit of using this SAS® statement, is that it will run all possible model combinations of the given independent variables, and then produce a listing of those models based on the best adjusted coefficient of multiple determination.

"The advantage of using regression is that it can show whether a model's estimates correlate well with experience even when the MRE test does not" (Kemerer, 1987: 421). Kemerer used linear regression on an uncalibrated COCOMO 1981, and found that the R^2 value for the Detailed model was 52.5 percent with a resultant regression equation as follows (Kemerer, 1987: 423):

$$\text{Actual Man Months} = 66.8 + 0.118 * (\text{COCOMO}_{est}) \qquad (16)$$

However, this equation is suspect since it was not validated for basic assumptions of regression, such as normality and equal variance (Matson, et al., 1994:278-280). The initial regression model for this study will take the following linear form:

$$Y = \beta_0 + \beta_1 X_i + \varepsilon \qquad (17)$$

where: Y = calculated effort (dependent variable);
β_0 = constant (the y-intercept);
β_1 = dependent variable coefficient or slope of the best-fit line;
X_i = COCOMO II effort estimate;
ε = error term = residual.

The following assumptions of the data will apply for this univariate, regression analysis:

1. There is a constant variance for the error terms, i.e. assume the data is homoschedastic.
2. The error terms are normally distributed about the true regression surface.
3. The model is linear as specified.
4. The causation system (independent variables) is constant and will remain constant.

Normality and homoschedasticity will be tested by visually inspecting residual plots and X Y Scatter plots. According to D'Agostino and Stephens, there is no test that is optimal for testing normality in all cases, but in most cases, the most powerful method for testing for normality is the Shapiro-Wilk test (D'Agostino and Stephens, 1986:403). The results of the COCOMO II effort estimates will be tested for normality by entering the residual values into Statgraphics Plus for Windows®, Version 2, which produces the results of a Shapiro-Wilk test (Statgraphics Plus, 1995). This test is a method for determining skewness or direction of the model by analyzing the residuals (difference between the regressed COCOMO II effort estimate and actual effort) (D'Agostino and Stephens, 1986:403).

Homoschedasticity will be also tested by using an equal variance test. This test is easily executable within Microsoft Excel®. According to Magee, homoschedasticity may be tested by dividing the original data set into two subsets; the first subset representing the lowest effort projects, and the second subset representing all the highest effort projects (use actual effort to determine subsets) (Magee, 1986:123). The next step is to develop a least squares, best fit line to each subset. Then, divide the sum of the squared error term (from the ANOVA Table) of the higher effort subset by the sum of the squared error term of the lower effort subset; this value is equal to the calculated F-value

(Magee, 1986:123). Degrees of freedom may be derived by using the following equation (Magee, 1986:123):

$$df = T - k - 1 \qquad (18)$$

where: T = total number of data points in the subset

k = number of dependent variables

Using an F-Table from any statistics book, the hypothesis may be developed by determining the critical value based on the desired confidence level (in this case, 90 percent will be used throughout) and degrees of freedom.

H_0 (null hypothesis): Variance is characterized by heteroschedasticity.
$F_{calc} > F_{table}$
H_1 (alternative hypothesis): Variance cannot be shown to be characterized by homoschedasticity.
$F_{calc} \leq F_{table}$

Model significance will be analyzed using the F-test and t-test (use values from the ANOVA Table), and the explanatory power of the model will be analyzed using the coefficient of determination (R^2). Independent variable significance will be analyzed using the p-value at the 90 percent confidence level.

If the model fails to support the regression assumptions, then the equation will be transformed using SAS® to determine the best fit equation and the assumptions will be reanalyzed using the tests described previously. This transformation could result in a multivariate model. Since each variable is mathematically related to the first variable, estimated person months, multiple correlation will be introduced into the model. However, in this situation, it's important to recognize that this will only affect the researcher's ability to determine the effects of the individual coefficients (β) on the model. Therefore, it is possible that all the assumptions will not be met. If so, the best fit equation will be identified and limitations discussed. A multiplicative model will only be attempted if the SAS® statement (see page 47) given above does not produce an improved regression model than originally produced, and if indicated by analysis. In most cases, anomalies will be difficult to ascertain since little is known about the data except for what is given in the SMC SWDB. Therefore, influential outlier tests such as DFBeta, Cook's D, etc. will not be run.

Overall, the hypothesis for this analysis will be tested at the 90 percent confidence level as follows:

H_0 (null hypothesis): Calibration does not improve the accuracy of the model based on using Conte's criteria.
H_1 (alternate hypothesis): Calibration does improve the accuracy of the model based on using Conte's criteria.

Once a best fit regression model is determined, then new effort estimates will be calculated, as well as corresponding MRE values, MMRE, and Pred(.25).

The data used in Kemerer's study was mainly language specific, COBOL in this case. It's expected that in linear form, the R^2 value for this research should be better than that reported by Kemerer. This is assuming that COCOMO II is a better and more up-to-date model then its predecessor. Like Kemerer's research, this research will also be based on using only estimated effort, as opposed to using schedule. Time will dictate to what detail, if at all, this portion of the research will be analyzed. The more time available, the greater the detail for the regression analysis.

Summary

The methodology used is assumed sound based on proven and accepted mathematical and statistical analysis. There are several known weaknesses with the database that may affect the outcome of this research. Once the data stratification begins, it is hoped that there will be enough data points in each data set to conduct a proper analysis. This would require a minimum of 12 data points per data set, using 10 of those points for calibration. In addition, and time permitting, this research will be expanded to include a regression analysis of the results.

IV. Findings

Overview

The objective of this chapter is to discuss the results of the research. First, the results of stratifying the SMC SWDB will be discussed and any adjustments to the data that were necessary to comply with the COCOMO II model. Next, the results of the calibrated and validated data sets will be discussed in terms of Conte's criteria and the Wilcoxon Signed-Rank Test. Lastly, the regression analysis results and best fit equations will be discussed.

SMC SWDB Stratification

The actual steps used in performing the SMC SWDB queries were originally discussed in Chapter II. Directly prior to stratifying the database, SMC SWDB 3.0 was received; however, there were difficulties in running queries on the database, and the research had to be conducted using version 2.1. The objective of stratifying the database was to isolate the queries down to homogeneous operating environments, and if possible, down to homogeneous applications for each data set. The reason it's important to stratify the database, is that the more homogeneous the operating environment and application (e.g. Command & Control), the better the software cost model should be able to more accurately estimate effort.

Initially, the query was setup as illustrated in Table 5; however, one adjustment to the queries was necessary, and several refinements to the query results were necessary. All fields in the SMC SWDB were examined by saving the results to a database file that could be opened in Microsoft Excel®. The report writer option found in SMC SWDB will not allow the user to analyze all the available fields in the database. The first adjustment was to eliminate European developments from all of the queries that resulted with U.S. and European developments. As stated in Chapter II, this was done to strengthen the estimates by the model by eliminating the possible inconsistencies found between the European and U.S. development methods (Ferens, 1997). European developments were found in Military Ground (Signal Processing), Ground in Support of Space, and the Unmanned Space categories. The next adjustment was to change the application parameter from 'All' to 'Command & Control' for the Ground in Support of Space query, to reduce the number of data points from 32 to 15, creating a homogenous data set and allowing easier manipulation of the data. Using the resampling method will improve the quality of the validation results, which alleviates the necessity for large data sets. The Missile query only resulted in 4 projects, and therefore, it will not be used in this study.

Each project is supplemented by 276 records (or fields) describing size, cost, development, contractor (although, the contractor is not identified), and program characteristics. Many of the fields were identified with a zero, blank, or negative one, to indicate a null entry. If this is the case for one of the necessary fields (e.g. Personnel Capabilities), then that entry will be considered nominal when entered into COCOMO II.

One of the critical fields for each project was entitled 'Confidence Level.' This field is not visible in the report writer option of SMC SWDB; it must be viewed from the complete database field listing, which may be done with Microsoft Excel®. The confidence level is a subjective parameter that estimates the confidence in the normalized size and effort data and is based on the amount and consistency of the new software size, pre-existing software size, percent re-design, percent re-code, percent re-test, and software development phases data that is provided (Southwell, 1996:35). "The confidence level is an indicator of how likely the SMC SWDB normalized data accurately represents the true normalized size and effort. Higher confidence levels represent normalized estimates based on complete and consistent data; lower confidence levels represent normalization estimates based on incomplete or inconsistent data" (Southwell, 1996:35). Projects with less than a nominal confidence level were eliminated from the data sets. This resulted in four projects being eliminated. Two projects were eliminated from Unmanned Space and one project was eliminated from both the Mil-Spec Avionics query and the Military Mobile query. Like the Missile query, Unmanned Space and Mil-Spec Avionics data sets were eliminated from the study for lack of data points (< 12). However, four queries (Military Ground - C^2, Military

Table 8. SMC SWDB Resultant Queries

Query Title	Operating Environment	Applications	Original # of Projects	Final # of Projects
Mil Grd - C^2	Mil Grd	Cmd & Cntrl	12	**12**
Mil Grd - SP	Mil Grd	Signal Proc	20	**19**
Unman Space	Unman Space	All	29	10
Grd in Suport of Space	Grd in Support of Space	Cmd & Cntrl	82	**15**
Mil Mobile	Mil Mobile	All	13	**12**
Missile	Missile	All	4	4
Mil-Spec Avion	Mil-Spec Avion	All	12	11

Ground - Signal Processing, Ground in Support of Space, and Military Mobile) ended up with ≥12 data points in which to conduct the calibration and validation on. Some details of the queries, changes, and number of projects is shown in Table 8, with those applications containing 12 or more projects highlighted. The resultant projects and corresponding data may be seen in Appendix B.

Analysis of SMC SWDB Fields to COCOMO II. The next step in the data stratification process was to determine which fields from the SMC SWDB could be applied to the factors in COCOMO II. The only factors of concern for this study were the Effort Adjustment Factors (EAFs). Analysis of the available SMC SWDB fields verified that the Scaling Factors (SFs) were not represented by the fields within the database. The

EAFs and similar SMC SWDB fields with any proposed adjustments to ratings are discussed below.

1. RELY (Required Software Reliability) "...is the measure of the extent to which the software must perform its intended function over a period of time" (USC, COCOMO II Model Definition Manual, 1997:35). The Quality Assurance Level field "...is usually directly related to the impact that a failure in the software would have during its operational phase (MCR & Cost Management Systems (CMS), 1995:B-25). The rating levels between these two attributes will be assumed the same; therefore, no adjustments are necessary.
2. DATA (Data Base Size)"...attempts to capture the affect large data requirements have on product development" (USC, COCOMO II Model Definition Manual, 1997:35). To determine the effective DATA rating, the database size and program size are required. Database size was only given in a couple of instances for the queried projects, therefore this will be assumed nominal for all cases.
3. DOCU (Documentation match to life-cycle needs) "...is evaluated in terms of the suitability of the project's documentation to its life-cycle needs" (USC, COCOMO II Model Definition Manual, 1997:36). There is no match to this attribute in the SMC SWDB; therefore, DOCU will be set to nominal for all projects.
4. CPLX (Product Complexity) can be a subjective rating based on the combined effects of control operations, computational operations, device-dependent operations, data management operations, and user interface management operations (USC, COCOMO II Model Definition Manual, 1997:35-36). The rating may be subjective due to the five effects and categorization of those effects based on a table which is given on page 41 of the COCOMO II Model Definition Manual to help determine the rating. CPLX is best represented by the Inherent Difficulty of Application (APPL DIFF) parameter found in the SMC SWDB. The APPL DIFF is a rating of the "...complexity of the

software development independent of the developer's ability to implement the component (MCR & Cost Management Systems" (CMS), 1995:B-23). No adjustments will be necessary of the APPL DIFF rating scale, since it seems to parallel that of the CPLX rating scale.

5. RUSE (Required Reusability) "...accounts for the additional effort needed to construct components intended for reuse on the current or future projects" (USC, COCOMO II Model Definition Manual, 1997:36). The best match to this attribute in the SMC SWDB is the Reusability Requirements (REUSE REQM) field which identifies the level of reusability (MCR & CMS, 1995:B-13). However, the rating levels are not substitutable as they are and must be adjusted as follows in Table 9 below. Based on matching the two rating systems as closely as possible, note that the High rating will not be utilized. The Very High rating could have been chosen not to be used, but it's felt that as long as everything is consistent within the identification process, that this should not make a difference which rating level is ignored.

Table 9. Reuse Requirements

RATING	REUSE REQM	RUSE	Adjusted Rating
Very Low	n/a	n/a	n/a
Low	n/a	none	Nominal to Low
Nominal	no reusability reqmt	across project	High to Nominal
High	reusability desired	across program	not used
Very High	developed exclusively for reuse	across product line	Very High to Very High
Extra High	full reusability required	across multiple product lines	Extra High to Extra High

(USC, COCOMO II Model Definition Manual, 1997:36) and (MCR & CMS, 1995:B-13)

6. TIME (Execution Time Constraint) "...is a measure of the execution time constraint imposed upon a software system. The rating is expressed in terms of the percentage of available execution time expected to be used by the system..." (USC, COCOMO II Model Definition Manual, 1997:36). Again, there is no adequate record in the SMC

SWDB to equate to the TIME factor; therefore, it will be set to nominal for all projects.

7. STOR (Main Storage Constraint) "...represents the degree of main storage constraint imposed on a software system or subsystem" (USC, COCOMO II Model Definition Manual, 1997:37). STOR also has no sister factor within the SMC SWDB; therefore, like TIME, it will be set to nominal. Anyhow, according to Boehm, it's questionable as to the importance of this factor, because of the "...remarkable increase in available processor execution time and main storage subsystem" (USC, COCOMO II Model Definition Manual, 1997:37).
8. PVOL (Platform Volatility) is rated based on the number of major changes to the platform. "Platform is used here to mean the complex of hardware and software the software product calls on to perform its tasks" (USC, COCOMO II Model Definition Manual, 1997:37). There is no factor in the SMC SWDB that directly matches the PVOL factor; therefore, it will be set to nominal for all projects. However, there is one SMC SWDB parameter that is similar, the Development System Volatility factor, which will be discussed in Item 18 as a USER 1 parameter.
9. ACAP (Analyst Capability) is rated based on the ability of the analysts to design and work on requirements, in addition to their efficiency, thoroughness, and social (communication and cooperation) abilities, and not their experience level (USC, COCOMO II Model Definition Manual, 1997:37). The SMC SWDB has one parameter, Personnel Capability (PERS CAP), that encompasses both analyst and programmer capabilities. PERS CAP will be used to determine the ACAP parameter; adjustments to the SMC SWDB ratings will not be necessary.
10. AEXP (Analyst Experience) is determined by the project team's application experience (USC, COCOMO II Model Definition Manual, 1997:38). AEXP is similar to the Personnel Experience (PERS EXP) factor found in the SMC SWDB.

Although AEXP is based on actual months of experience and PERS EXP is defined as 'experts', 'gurus', etc., the ratings seem highly similar and will be used as is.

11. PCAP (Programmer Capability) is a rating of the programmer team's ability, efficiency, thoroughness, and social (cooperation and coordination) ability (USC, COCOMO II Model Definition Manual, 1997:38). As with ACAP, PERS CAP will be used to determine the PCAP parameter. Adjustments to the SMC SWDB ratings will not be necessary.
12. PEXP (Platform Experience) represents the programmer's understanding of more powerful platforms, such as: graphic user interface (GUI), database, networking, and distributed middleware capabilities (USC, COCOMO II Model Definition Manual, 1997:38). The SMC SWDB does not contain any fields that parallel the PEXP parameter; therefore, PEXP will be set to nominal for all projects.
13. LTEX (Language Team Experience) "...is a measure of the level of programming language and software tool experience of the project team developing the software system of subsystem" (USC, COCOMO II Model Definition Manual, 1997:38). LTEX is actually a combination of two SMC SWDB fields, Team Programming Language Experience (TEAM LANG) and Development Methods Experience (DEV METH EXP). The rating scales are not exactly the same, which will require adjustment as shown in Table 10 on the following page. If the TEAM LANG and DEV METH EXP do not agree on the same rating after adjustment, then the lowest rating of the two fields will be used.

Table 10. Team Language and Tool Experience

RATING	TEAM LANG and DEV METH EXP	LTEX	Adjusted Rating
Very Low	< 4 months exp	≤ 2 months exp	Very Low to Very Low
Low	4 months average exp	6 months exp	Low to Low
Nominal	1 year average exp	1 year exp	Nominal to Nominal
High	2 years average exp	3 years exp	High to Nominal
Very High	3 years average exp	6 years exp	Very High to High
Extra High	≥ 4 years average exp	n/a	Extra High to Very High

(USC, COCOMO II Model Definition Manual, 1997:38) and (MCR & CMS, 1995:B-20)

14. PCON (Personnel Continuity) is a rating of annual personnel turnover (USC, COCOMO II Model Definition Manual, 1997:39). This parameter cannot be determined from any of the fields in the SMC SWDB; therefore, it will set to nominal for all projects.
15. TOOL (Use of Software Tools) is a rating of the type of tools used during development from simple edit, code, and debugging up to strong, mature, life-cycle tools (USC, COCOMO II Model Definition Manual, 1997:39). TOOL is very similar to the Automated Tool Support (AUTO TOOLS) found in the SMC SWDB. The ratings rank from Very Low to Very High for both factors, and will not require any adjustment.
16. SITE (Multisite Development) is rated based on the assessment and average of site collocation and communication support (USC, COCOMO II Model Definition Manual, 1997:39). There is a Multiple Site Development field in the SMC SWDB, but it does not reflect the same information as required by SITE. Therefore, SITE will be set to nominal for all projects.
17. SCED (Required Development Schedule) identifies the schedule constraint imposed upon the project (USC, COCOMO II Model Definition Manual, 1997:39). This

parameter will also be set to nominal for all projects, since there are no known fields in the SMC SWDB that equate to the SCED parameter.

18. USER 1 will be identified as Development Systems Volatility (VOLAT). This parameter will capture the "...difficulty that was caused by changes to the virtual machine.... Each change may (have) cause(d) developers to lose time due to learning the system, changing their code, procedures, etc." (MCR & CMS, 1995:B-24). VOLAT will be used in place of PVOL, which was set to nominal because no fields in the SMC SWDB could be used to define PVOL. The ratings and values assigned will be the same as PVOL, which will require some adjustments to the VOLAT ratings as described in Table 11 on the following page.
19. USER 2 will not be used.

Table 11. USER 2 - Development Volatility Rating

RATING	VOLAT	PVOL	Adjusted Rating
Very Low	n/a	n/a	n/a
Low	Essentially no changes	major change every 12 months	Low to Low
Nominal	Small non-critical changes	major change every 6 months	Nominal to Low
High	Occasional moderate changes	major change every 2 months	High to Nominal
Very High	Frequent moderate and occasional major changes	major change every 2 weeks	Very High to High
Extra High	Frequent moderate and frequent major changes	n/a	Extra High to Very High

(USC, COCOMO II Model Definition Manual, 1997:36) and (MCR & CMS, 1995:B-24)

Once the SMC SWDB fields, as well as necessary adjustments for normalization, were identified, the next step was to calculate the Effort Multiplier (EM). The EM is simply equal to the multiplication of all the EAFs within one project. Tables in Appendix B show the results of the EMs, one for each of the four applications, as well as the adjusted

EAFs that were determined for each parameter within each project. The final step before calculating the coefficient 'A' for each query was to determine any anomalies among the projects within the same applications.

The approach used to identify anomalies was to compare the productivity rate and effort multiplier (EM) between the various projects within each application. Those projects that demonstrated a high or low productivity rate when compared to other projects and their given EMs were eliminated. This step is very subjective; therefore, only data that demonstrated what appeared to be extreme productivity and EM values were eliminated. Data was not eliminated with the purpose to improve the accuracy of the model, but was eliminated because it appeared to be a bona fide outlier. Hence, an analyst with software project estimation experience could play a critical role at this point. An analyst's understanding and experience with past projects could help determine at what point to eliminate and keep given projects within a data set, or research key information that may be missing which could result in keeping data or improving its value to the calibration. The productivity rate was calculated for each project as follows:

Productivity Rate = (Size in KSLOC) / Actual Effort (16)

Actual Effort was adjusted as necessary for missing phases, the results of which may be seen in four individual tables in Appendix B. These tables also show all the resultant calculations for the EMs and productivity rates for each project. After considering the value of each project and comparing it to the other projects in the database, it was determined to only eliminate 3 projects (projects 1, 3, and 13—see Table entitled "Ground in Support of Space, Productivity/EM Comparison") from the Military Ground - Signal Processing application because they were characterized by extreme productivity rates in comparison to their individual EM. It's imperative to understand that this portion of the data stratification process is extremely subjective and many more projects could have been eliminated. However, after consideration, it was determined that without greater insight to the specifics of each project and without sufficient experience in these applications, it was best to leave the remaining data points within the data sets. Elimination of any data points that display even a slight deviation from the norm, would most surely bias the data and create a false representation of the accuracy of COCOMO II.

Calibrating the Coefficient. Microsoft Excel® Version 7, Random Number Generator was used to divide the data sets into a calibrated and validated subset. The results of using the random number generator are listed in tables in Appendix C1. Once the subsets were determined for each application, the next step was to calculate the coefficient using the steps from Chapter III. Equations 8a and 10 were used directly from Chapter III to calculate the optimal coefficient, A. Tables in Appendix C1 show the results of each resampling run (five for each data set) that was used to calculate A. After investigation of the coefficients, some concern as to the range of the coefficients was raised. Table 12 below reveals that the coefficient range is quite significant. The coefficient range for Military Mobile indicates that there will be a

Table 12. Range of 'A' Coefficient

APPLICATION TYPE	$A_{default}$	A_1	A_2	A_3	A_4	A_5	A Range	A Avg	Range %
Mil Grd--C^2	2.45	2.0513	1.9758	2.6025	1.9942	1.8647	0.7378	2.0977	30.114
Mil Grd--Signal Proc	2.45	3.3321	2.8300	3.2509	3.2331	3.2564	0.5021	3.1805	20.494
Grd in Support of Space	2.45	1.8077	1.1807	1.3579	1.5547	1.3792	0.6270	1.4560	25.592
Mil Mobile	2.45	4.4460	7.5260	7.2211	7.5207	7.5860	3.1400	6.8600	128.163

wide range on the effort estimates and reduces the probability of attaining Conte's preset criteria for that set of data. It's assumed that accurately reported, as well as consistently reported data should result in a small coefficient range. Table 12 indicates that the effort estimates will vary from actual estimates. For example, since the only difference between each calibrated run for each project is the coefficient, a coefficient that varies by

30 percent indicates there will also be a 30 percent estimated effort range (± 15 percent) between the five separate runs for a single project.

For a Cost Analyst or organization wishing to calibrate the model to a specific coefficient, this may be accomplished by taking an average of the coefficients for each particular data set and substituting into Equation 3 (in COCOMO under the calibration pull-down window) for the coefficient A. Table 12 above, includes the average coefficient derived for each data set.

Calibration & Validation Results

As indicated in Chapter III, COCOMO II was initially run in the default mode (A = 2.45) and estimates were generated for both calibrated and validated subsets. Likewise, COCOMO II was calibrated and then each calibrated model was used to generate estimates for both calibrated and validated subsets. Ideally, in default mode, there should not be any significant difference between MMRE, RRMS, and Pred(.25) for the calibrated and validated subsets within each application data set. However, when the model is run in the calibrated mode, the MMRE, RRMS, and Pred(.25) should be superior for the calibrated subset versus the validated subset, since the calibrated subset was used to actually calibrate the model. The estimated person months of effort for each run and the corresponding actual effort are shown in Appendix C2 (one table per data set). Besides calculating a single effort estimate, COCOMO II also produces an optimistic and pessimistic estimate. Neither of these estimates were used since they are simply based on the assumption that an optimistic estimate is 80 percent of the calculated estimate (termed most likely) and the pessimistic estimate is 125 percent of the most likely estimate. In the tables in Appendix C2, the optimistic and pessimistic estimates are given for the default mode only and intended strictly for a cursory look by the reader if desired.

One method to help in quickly assessing whether the model produces effort estimates patterned after the actual effort is to examine charts. In Figures 2, 3, 4, and 5, the default estimate is plotted against the actual effort. In all cases, the default estimates do seem to coincide with the actual effort. It's also notable that in most cases, the line charts also show that the estimates seem to be less than the actual effort. The Wilcoxon Test, which helps to determine estimating bias, will be discussed later and will help to highlight model biases. Although these charts can be used to visually ascertain the given situation, further analysis (equal variance and Shapiro-Wilk tests) will be necessary to derive a sound analysis.

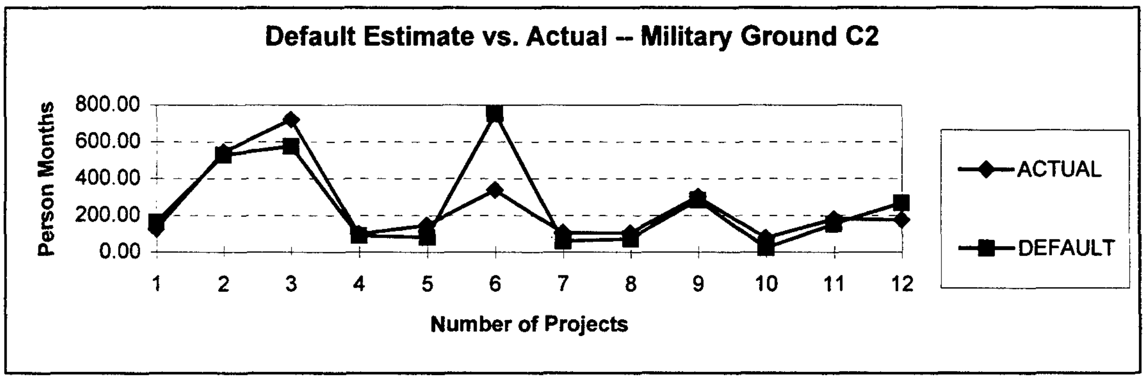

Figure 2. Default Estimate vs. Actual Effort—Military Ground, C^2

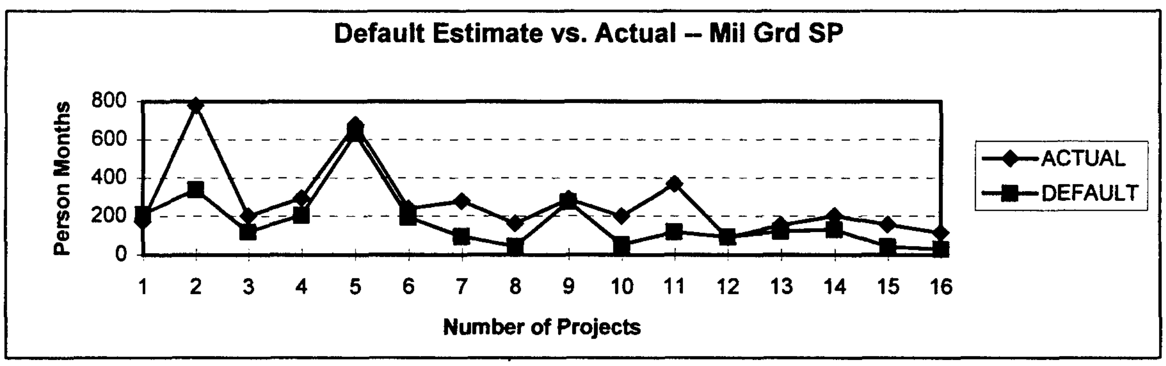

Figure 3. Default Estimate vs. Actual Effort—Military Ground, Signal Processing

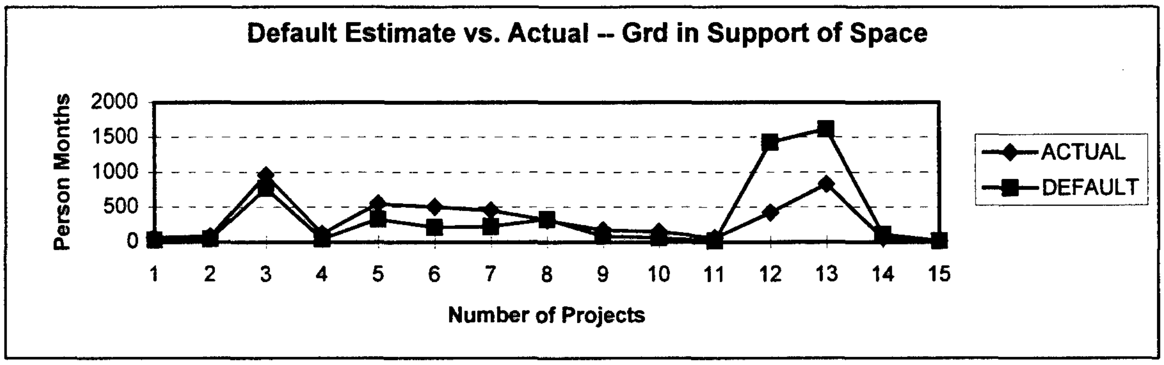

Figure 4. Default Estimate vs. Actual Effort—Ground in Support of Space

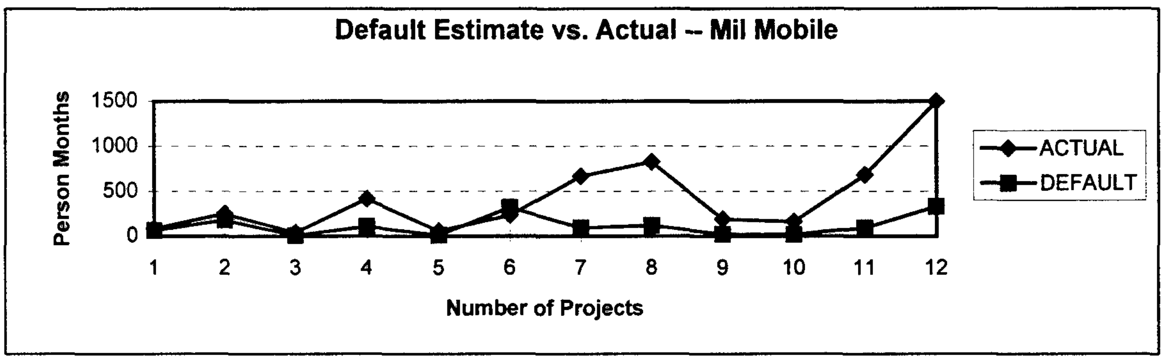

Figure 5. Default Estimate vs. Actual Effort—Military Mobile

For Figure 2, note that the sixth project appears to be an outlier. Further analysis of this project shows that the productivity rate is the second highest in the data set, yet the EM seems to be the similar to other projects with lower productivity rates. This would explain the large error between the actual and estimated effort. MRE and MMRE will be used to further analyze each data point and each data set. For Figure 3, the second project (which is actually project 4 in the data set since projects 1 and 3 were eliminated) appears to be an outlier. Further analysis of this project indicates a low productivity rate which could be due to program complexity, poor management, schedule stretch out, etc.; all of which is beyond the ability of this researcher to determine. The other figures also show outliers due to differences in productivity rates. However, it's impossible at this point to determine the reasons for the anomalies. The regression analysis, which is discussed in the next section, will help in determining accuracy and bias of the COCOMO model.

Charts like Figures 2 thru 5 were not produced for the calibration estimates; however, the range of estimates for each project within a given application were calculated. To illustrate the possible impact of this range, the range was expressed into terms of the actual effort with the following equation:

$$\text{Range \%} = (\text{Range} / \text{Actual Effort}) * 100 \qquad (18)$$

This, more so than the charts above, illustrates mathematically how tight or loose the estimates could be from the actual effort within each application. Tables 13, 14, 15, and 16 on the following pages, show the results of the ranges for each application with the highest and lowest ranges shown in bold face type. In the first column of each table, the calibrated effort range is given. In the next column, the calibrated effort range is put into terms of the actual effort. This gives the analyst the ability to compare between projects. For example, the first range given in Table 13 is 49.34, which is much less than the second range of 159.25. However, as a percentage of the actual effort, the second range only varies by 29 percent, whereas the first range varies by 39 percent. Each table also lists MRE for each model run. As discussed in Chapter III, the model was calibrated five times and then ran against each data point within the validated and calibrated subsets (all the data points). The MRE shows absolute value of the difference between the calibrated and actual efforts as a percentage of the actual effort. According to Conte, this value is best if it is equal to or less than 25 percent (Conte et al., 1986:172-175).

For Military Ground—C^2, the calculated range is between 10 and 67 percent. Compared to the other applications, this is average. As Figure 1 showed above, the sixth project has the greatest range and could possibly indicate an outlier, which if eliminated, could improve the model's ability to produce accurate estimates. Nonetheless, for the same reasons as stated earlier, this project was not eliminated. The MRE indicates that for the calibrated range, the best estimate was within 0.24 percent of the actual effort, while the worse case was 135.23 percent. The worst case estimate is based on project six.

It's important to note, that even though a specific project may have a correspondingly low percentage range, the actual MRE can be quite high. For example, the project with the lowest range percentage (9.56), has MREs between 66 and 76 percent

For Military Ground—Signal Processing, the calculated ranges and individual MREs are much tighter than for the command and control application, or any of the other applications. This indicates that this application should have the best estimating results, because the actual effort, productivity rate, and EAFs, appear to be more consistent. On the downside, and in terms of the EAFs, this particular data set was characterized by the least amount of reported data of any of the data sets. Except for project 1 (which was eliminated earlier based on a high EM), all other EAFs were assumed nominal. In this particular case and to this point of the research, it seems to be advantageous that all the EAFs are recorded as nominal. If the MMRE, RRMS, and Pred(.25) result in being the best for this data set, then this could indicate that when the application is the same and all productivity rates are within reason (remember, 3 projects were eliminated for inconsistent productivity and EM rates), then it may be advantageous to set all the factors to nominal and calculate the coefficient based on an EM of one.

For Ground in Support of Space, the calculated ranges and individual MREs are not very good. It seems that the calibration of the coefficient was driven by projects 12, 13, and 14, which is supported by Figure 3 and the table entitled "Ground in Support of Space, Productivity/EM Comparison in Appendix B. The productivity rate seems to be slightly high (however, in the opinion of this researcher, not extreme) when compared to the EMs and other projects. Investigation of the original data base shows that for projects 12 and 13, very little data was reported and the parameters had to be assumed as nominal. Again, a lack of consistent data could be a significant factor.

The last application, Military Mobile, has a widely varying set of ranges and very poor MRE values. The range percentage spans from 14 to 175 percent and the MREs are the worse yet. Thirteen of the MREs register over 100 percent, and four of those are over 300 percent. These poor values could indicate a key parameter is missing from the model, or the data is incorrect and erroneous. Looking back at the original data, project six seems to contradict itself. One of the fields available in the SMC SWDB is software complexity. Out of all the projects, it is the only one listed as difficult; however, this project has the highest productivity rate. Except for the first two projects which used Assembler language, the other ten projects all used Ada; therefore, the language used does not seem to be significant in this case. Another parameter to consider is the application type. Military Mobile includes all application types from data base to mission planning, command and control, and signal processing. The EMs also appear to vary a lot, but it is difficult to eliminate data points when there are a myriad of applications being estimated. The application type could be a key to estimating effort, because all the other queries were isolated to one application (command and control or signal processing). The Military Mobile data set includes all applications and, not surprisingly, has the worse MRE values, which supports the assumption that calibration using a homogenous data set is key to developing more accurate estimates. Applying regression to this data set will lead to greater insights to this assumption.

Table 13. Military Ground--C^2, Range Percentages of Effort

	PM_{range}/PM_{actual}	MRE in %				
PM_{CAL} Range	%	Run 1	Run 2	Run 3	Run 4	Run 5
49.34	38.97	8.35	4.36	37.47	5.33	1.51
159.25	29.20	18.83	21.81	2.99	21.08	26.21
173.81	24.09	33.03	35.50	15.04	34.90	39.13
27.28	27.22	24.32	27.10	3.98	26.43	31.20
24.91	16.99	52.77	54.51	40.08	54.09	57.07
226.54	**66.69**	85.41	78.59	**135.23**	80.25	68.54
18.63	17.48	51.40	53.19	38.34	52.75	55.82
22.28	21.12	41.28	43.44	25.50	42.91	46.62
85.75	28.42	20.99	23.90	**0.24**	23.19	28.18
7.47	**9.56**	73.41	74.39	66.27	74.15	75.83
46.24	25.48	29.15	31.76	10.11	31.12	35.59
80.82	45.87	27.54	22.85	61.81	23.99	15.94

Table 14. Military Ground—Signal Processing, Range Percentages of Effort

	PM_{range}/PM_{actual}	MRE in %				
PM_{CAL} Range	%	Run 1	Run 2	Run 3	Run 4	Run 5
43.54	**25.01**	65.99	40.98	61.95	61.06	62.22
69.38	8.91	40.86	49.77	42.30	42.62	42.20
24.52	12.10	19.68	31.78	21.63	22.06	21.50
42.11	14.36	**4.72**	19.07	7.04	7.55	6.88
129.82	19.08	26.60	7.53	23.52	22.84	23.73
39.96	16.61	10.25	6.36	7.57	6.98	7.75
19.40	6.96	53.79	60.75	54.91	55.16	54.84
8.91	5.49	63.59	69.07	64.47	64.67	64.41
56.62	19.59	29.98	10.39	26.81	26.11	27.02
10.86	5.42	64.03	69.45	64.91	65.10	64.85
24.16	6.58	56.32	62.90	57.39	57.62	57.32
19.32	21.29	41.29	20.00	37.85	37.09	38.08
25.15	16.44	9.12	7.32	6.46	5.88	6.64
27.03	13.34	11.45	24.79	13.61	14.08	13.46
8.42	**5.36**	64.46	**69.81**	65.32	65.51	65.27
6.30	5.48	63.66	69.14	64.55	64.74	64.49

Table 15. Ground in Support of Space, Range Percentages of Effort

	PM_{range}/PM_{actual}	MRE in %				
PM_{CAL} Range	%	Run 1	Run 2	Run 3	Run 4	Run 5
8.72	13.55	60.94	74.49	70.66	66.41	70.20
12.30	14.58	57.97	72.55	68.43	63.85	67.93
197.43	20.52	40.84	61.36	55.56	49.12	54.86
11.96	9.86	71.57	81.43	78.65	75.55	78.31
84.94	15.39	55.62	71.01	66.66	61.83	66.14
54.86	10.88	68.63	79.51	76.44	73.02	76.07
57.49	12.61	63.63	76.25	72.68	68.72	72.25
83.40	26.71	23.00	49.71	42.16	33.78	41.26
23.21	13.41	61.33	74.74	70.95	66.74	70.50
15.68	10.62	69.39	80.01	77.01	73.68	76.65
5.04	**8.37**	75.86	84.23	81.86	79.24	81.58
364.83	**86.24**	**148.63**	62.39	86.77	113.83	89.69
413.17	49.76	43.47	6.29	7.77	23.39	9.46
28.61	47.58	37.18	10.40	**3.04**	17.98	4.66
5.38	28.34	18.29	46.63	38.62	29.73	37.66

Table 16. Military Mobile, Range Percentages of Effort

	PM_{range}/PM_{actual}	MRE in %				
PM_{CAL} Range	%	Run 1	Run 2	Run 3	Run 4	Run 5
84.30	96.27	36.32	130.75	121.40	130.59	132.59
229.15	91.65	29.76	119.66	110.76	119.50	121.41
10.72	26.06	63.10	37.53	40.06	37.58	37.03
139.96	33.50	52.57	**19.70**	22.96	19.76	19.06
12.30	20.82	70.52	50.10	52.12	50.14	49.70
407.41	**174.74**	147.42	318.82	301.85	318.52	**322.16**
118.28	17.71	74.92	57.55	59.27	57.58	57.21
151.19	18.30	74.08	56.13	57.91	56.16	55.78
27.69	14.58	79.35	65.05	66.47	65.07	64.77
22.90	**14.28**	79.78	65.77	67.16	65.79	65.50
121.28	17.77	74.84	57.41	59.14	57.44	57.07
425.04	28.41	59.77	31.90	34.66	31.95	31.36

Table 17. Wilcoxon Test Results

	Default Mode			Calibrated Mode		
APPLICATION TYPE	**T^+**	**T^-**	**Bias**	**T^+**	**T^-**	**Bias**
Mil Grd--C^2	4	6	Under	4	6	Under
Mil Grd--Signal Proc	2	13	Under	5	10	Under
Grd in Support of Space	4	11	Under	4	11	Under
Mil Mobile	0	10	Under	1	9	Under

The next step was to determine whether the estimates tended to be biased or not. This was done using the Wilcoxon Signed-Rank test. The results of this test are shown in Table 17 above. Under the T^+ and T^- headings, a number is listed, which indicates the number of estimates that were either below or above the actual effort. This test indicates that the COCOMO model tends to underestimate effort. With calibration, there was slight improvement in the bias, however, the trend was still to overwhelmingly underestimate.

Tables 18 and 19 below, present the results of the MMRE, RRMS, and Pred(.25) calculations. It was expected that the criterian should improve with calibration. In all cases with the validation subset, the calibrated model showed improvement. Surprisingly though, when analyzing the MMRE and Pred(.25) for the calibrated and default subsets, the prediction level worsened in all four cases, and the MMRE was worse in two cases, and only showed slight improvement for the other two. This was totally unexpected, because the model, if anything, should have showed improvement when checked against the data that was used to calibrate it. An explanation for this can be seen when the individual estimates are compared. It can be shown, that the calibrated model tended to be more accurate in estimating the higher effort projects, but, in turn, it gave up accuracy on all the other projects. This is further supported by the RRMS, which did show improvement in three of the four cases with the validation subset, and all four cases of the calibration subsets.

Table 18. Results of Default Model Accuracy

	DEFAULT ACCURACY					
	Calibration Data Set			*Validation Data Set*		
APPLICATION TYPE	MMRE	RRMS	PRED(.25)	MMRE	RRMS	PRED(.25)
Mil Grd--C^2	0.3598	0.5216	0.4400	0.3933	0.4858	0.3000
Mil Grd--Signal Proc	0.4084	0.4999	0.3846	0.4507	0.6275	0.3333
Grd in Support of Space	0.5941	1.1072	0.2167	0.7077	1.1652	0.0667
Mil Mobile	0.6817	1.0699	0.0800	0.7930	0.9458	0.1000

Table 19. Results of Calibrated Model Accuracy

	CALIBRATED ACCURACY					
	Calibration Data Set			*Validation Data Set*		
APPLICATION TYPE	MMRE	RRMS	PRED(.25)	MMRE	RRMS	PRED(.25)
Mil Grd--C^2	0.4037	0.4286	0.3000	0.3332	0.5318	0.4000
Mil Grd--Signal Proc	0.3890	0.4416	0.3692	0.3845	0.5343	0.4000
Grd in Support of Space	0.5885	0.7355	0.1167	0.6587	0.9498	0.2000
Mil Mobile	0.7030	0.8231	0.0800	0.6762	0.7381	0.0000

Regression Analysis

The purpose of conducting a regression analysis was to reveal if the default estimate accuracy could be improved using a best fit, least squares regression equation. The steps are outlined in Chapter III and were followed accordingly. The initial regression, a simple univariate model, was performed on each data set using the actual effort and default effort estimates using Microsoft Excel®, Version 7. The default effort estimates for each project within a data set were identified as the independent variables, whereas, the corresponding actual effort was identified as the dependent variable. The individual Excel outputs yielded the following four equations (standard error is shown in parenthesis below the coefficient and intercept):

$$PM_{milgrdc2} = 67.83629 + 0.690795(PM_{default})$$
$$(53.2) \qquad (0.155)$$
$$PM_{milgrdsp} = 101.2164 + 1.020515(PM_{default})$$
$$(45.3) \qquad (0.202)$$
$$PM_{grdsupsp} = 169.9644 + 0.41564(PM_{default})$$
$$(67.9) \qquad (0.111)$$
$$PM_{milmob} = 161.7233 + 2.321859(PM_{default})$$
$$(151) \qquad (0.972)$$

Figure 6. Default Regression Equations

Since the intercepts and beta coefficients are all positive, this supports the Wilcoxon Signed-Rank test that COCOMO II tends to underestimate. It could be that the model was actually designed to underestimate rather than overestimate. The impression this researcher developed when discussing the model with Dr. Boehm was commensurate with the idea that it's better to underestimate than overestimate, so as not to encourage poor management or a lack of motivation by the development team. Table 20 shows the results of the analysis of each regression model. The R^2, F, t, and p values were taken from the ANOVA tables. The next two columns (Resids Normal and X-Y Plot Linear) were determined based on visual inspection. The Shapiro-Wilk and Equal Variance tests are the results of a statistical analysis. As expected, Military Ground—Signal Processing had the overall best results, which reflects the tighter calibration range percentage discussed on page 65. Except for the Military Mobile application, each of the

Table 20. Initial Regression Run Results

Application	R^2	F	t-value	p-value	Resids Normal?	X-Y Plot Linear?	Shapiro-Wilk	Equal Variance?
Milgrdc2	.664	19.8	4.45	.001	no	no	no	no
Milgrdsp	.645	25.5	5.05	.0002	no	no	no	no
Grdsusp	.518	14.0	3.73	.003	no	no	no	no
Milmob	.363	5.7	2.39	.04	no	no	no	no

applications had solid F, t, and p-values. None of the models' residuals were normally distributed nor were the X-Y scatter plots linear as specified. However, the Military Ground—Signal Processing data set was extremely close to being normal based on visual inspection, although the Shapiro-Wilk test rejected that the plot was normally distributed. Due to the few data points within each data set, it was difficult to determine a specific pattern from the plots. Nonetheless, an attempt was made to visually discern a pattern from the plots.

Military Ground—C^2 is heteroschedastic (see definition in glossary) and appears parabolic at the same time. This type of relationship would infer a reciprocal or square root transformation. The heteroschedasticity implies a square root or log-log transformation, which indicates a level percentage increase as the independent variable increases by a certain value. Military Ground—Signal Processing appeared somewhat linear, and the residuals were very close to being normally distributed; however, what couldn't be determined by visual inspection, the Shapiro-Wilk test rejected for normality. The Ground in Support of Space model appeared to have either a parabolic relationship or log-log relationship. Based on this type of relationship, a square root function will be applied first, and if that doesn't prove satisfactory, and as a last resort, a log-log function will be applied. Lastly, it was clear that the Military Mobile data set displayed heteroschedastic characteristics.

Typically, heteroschedasticity can be eliminated by applying a log-log transformation, or less frequently, a square root function. Initially, it was hoped that the log-log transformation would not be necessary since it is more complex to evaluate and understand. The use of log-log models (multiplicative models) is prohibitive because of their difficulty by some to understand; however, it appears at this point, that the log-log transformation may be a viable and logical solution. All models were determined to be characterized by unequal variance (heteroschedastic) by the Equal Variance test described in Chapter III.

The next step was to run each model in SAS® and determine the best model based on the adjusted coefficient of multiple determination (R^2_{adj}). This step is only for informational purposes and was not intended to replace a logical process for determining a proper transformation. The program, recreated on the following page in Figure 7, was used for each application and only required a data set reference change. To run this model, a data set in SAS® containing $PM_{default}$, $PM^2_{default}$, $PM^{0.5}_{default}$, and $1/PM_{default}$, was created for each data set. The SAS® output generated for each regression run is shown in tables in Appendix D. The combination of variables that make up each model are determined by the highest R^2_{adj}. For this analysis, the model with the highest R^2_{adj} was looked at first by determining whether the listed variables appeared to be logical transformations of the original model based on the residual and X-Y plots.

For Military Ground—C^2, it was difficult to ascertain a definite pattern from the plots; therefore, as a starting point, all the independent variables were used for the regression run since this was the model with the highest R^2_{adj}. Unfortunately, a military

```
* THESIS--REGRESSION ANALYSIS OF;
* MILGRD C2;
OPTIONS LINESIZE=72; OPTIONS NOCENTER;
DATA ONE;
INFILE MILGRDC2;
INPUT PROJ PMACT PMDEF DEFSQ SQRTDEF RECDEF;
PROC REG;
      MODEL PMACT=PMDEF DEFSQ SQRTDEF
RECDEF/SELECTION=ADJRSQ;
PROC PRINT;
```

Figure 7. SAS Program

cost analyst is typically plagued by poor data or a lack of data; nevertheless, the analyst is still expected to do the best job possible with what data is available. The result of the regression run using all four variables in the model was extremely good. The resultant residual plots were normally distributed, and the X-Y plots were linear as specified. The results of the model are shown in Table 21 and Figure 8. To some, using all four variables appears to be data mining. In this case, since a relationship could not be determined and therefore, a logical transformation applied, the transformation was chosen based on the best adjusted coefficient of determination. Since the other three data sets required a log-log transformation, this was applied to the data set, but it was not normally distributed.

For Military Ground—Signal Processing, initially, it was visually determined that the model appeared normal in the first regression run using the default effort only. However, the Shapiro-Wilk and Equal Variance tests proved the initial impression wrong and rejected that the data set was normally distributed. The square root function was applied; however, it proved unsatisfactory. As a last resort, a log-log transformation was applied to the model and proved to be a successful transformation. In Table 21, it's shown that the Shapiro-Wilk test could not reject that the data set was normal; but, there was an adjustment made to this data set. When project 4 was included in the data set, the Shapiro-Wilk test result was to reject that the residuals were normally distributed. From visual inspection of the residual plot, project 4 appeared to be an outlier, and when project 4 was eliminated (see discussion for Figure 2), the Shapiro-Wilk test result was to not to reject that the residuals were normally distributed. In keeping with the assumptions, the log-log transformation of the Military Ground-Signal Processing data set yielded poorer F, t, and R values then the original regression run (compare Tables 44 and Table 49). However, by staying within the boundaries of the assumptions, the analyst can feel more confident about their estimate throughout the entire effort range (based on 2000 to 300000 SLOC), unlike what can be done with the estimate based on the original regression.

For Ground in Support of Space, the first attempt was to use the square root function; however, it proved non linear as specified and the residuals were not normally distributed. The log-log transformation was then applied to the model and resulted in a model that was linear as specified with normally distributed residuals, results of which are recorded in Table 21 and Figure 8. For Military Mobile (see table in Appendix D entitled "Military Mobile, SAS Regression Output"), the model listed first was the square root variable. The square root transformation was initially tried, but was non linear as specified and the residuals were not normally distributed. The log-log transformation was then applied to the variables and proved to be linear as specified with normally distributed residuals. The results of the this run were recorded in Table 21 and Figure 8. Tables containing the results of the Equal Variance and Shapiro-Wilk test are included in Appendix D.

When Kemerer did his study, in intermediate default mode using regression, the COCOMO 1981 model had a coefficient of determination of 0.599 (Kemerer, 1987:423). As stated before, Kemerer's results have been challenged by Matson, Barrett, and Mellichamp, since he did not test his assumptions (Matson et al., 1994:278-280). Nonetheless, his study was based on a language specific data set of 15 projects, 13 of which were Cobol based, 1 was Bliss, and the other was Natural (Kemerer, 1987: 421). In Table 20, we see that a similar coefficient of determination was achieved for each data set except for Military Mobile, which this researcher feels was not homogeneous enough to produce accurate results. Another similarity between the results of this study and Kemerer's results is that COCOMO tends to underestimate, this can be seen by comparing Equation 16 to Figure 6.

Table 21. Final Regression Run Results

Application	R^2	F	lowest t-value	p-value	Resids Normal?	X-Y Plot Linear?	Shapiro-Wilk	Equal Variance?
Milgrdc2	.919	19.8	-2.55	.038	yes	no	yes	yes
Milgrdsp	.540	16.5	4.06	.001	yes	yes	yes	yes
Grdsusp	.777	45.3	6.73	.00001	yes	yes	yes	yes
Milmob	.579	13.8	3.71	.004	yes	yes	yes	yes

$$PM_{milgrc2} = 2541.253 + 18.6776(PM_{def}) - 0.0093(PM_{def})^2 - 398.46(PM_{def})^{0.5} - 23418.1(1/PM_{def})$$

$$PM_{milgrsp} = EXP^{2.976707} * (PM_{def})^{.51157}$$

$$PM_{grdsusp} = EXP^{1.742717} * (PM_{def})^{.707902}$$

$$PM_{milmob} = EXP^{2.797758} * (PM_{def})^{.664101}$$

Figure 8. Final Regression Models

Once the best fit regression equations were determined for each data set, then the new effort was calculated using the applicable equation. Then, new MRE, MMRE, and Pred(.25) values were calculated based on the new effort. The results of each model is shown in Table 22, on the following page. Overall, the results were promising, with each data set showing an improved MMRE and Pred(.25) except for the Military Mobile data set, whose overall results displayed little change. The greatest change occurred with the Military Ground—C^2 data set, which met all of Conte's criteria. Although neither data set met Conte's criteria, both Military Ground—Signal Processing and Ground in Support of

Table 22. Accuracy Results of Final Regression Model

DEFAULT ACCURACY					Improved with Regression
	w/o Regression		*w/ Regression*		
APPLICATION TYPE	**MMRE**	**PRED(.25)**	**MMRE**	**PRED(.25)**	
Mil Grd--C^2	0.3671	0.4167	0.2059	0.7500	yes
Mil Grd--Signal Proc	0.4163	0.3750	0.3240	0.6250	yes
Grd in Support of Space	0.6168	0.2000	0.5140	0.3333	yes
Mil Mobile	0.7003	0.0833	0.7467	0.1667	no

Space had marked improvement, without applying much extra effort to do so. The improvement in MMRE and Pred(.25) for the Military Ground-Signal Processing is even more significant, because the log-log transform on this data set had lower ANOVA Table values than the default regression run. See final regression run tables in Appendix D for a listing of the independent variables, the newly calculated estimates using regression, and the associated MREs.

Summary

Although the results hoped for weren't achieved, it does appear that several points are worth mentioning. It was evident from the analysis that the COCOMO II model does tend to produce low estimates. The exact reason for this is unknown, but it is important to understand. It was also shown that, by applying regression to the COCOMO II results, the accuracy of the estimates can be improved. Further discussion of the results are presented in Chapter V.

V. Conclusions and Recommendations

Overview

The objective of this research was to calibrate the effort equation coefficient of the COCOMO II Software Cost and Schedule Model in the Post-Architecture mode to specific applications (i.e. Military Ground, Avionics, Unmanned Space) within the SMC Database, Version 2.1. The purpose of the calibration was to determine the accuracy (goodness of fit) of the model in default (uncalibrated) and calibrated modes, and validate the model's use by SMC and other DOD agencies to estimate program costs and schedules. The following criteria, as determined by Conte, Dunsmore, and Shen in their book Software Engineering Metrics and Models, was used to evaluate and validate the accuracy of the estimates: Mean Magnitude of Relative Error (MMRE) less than 0.25, Relative Root Mean Square (RRMS) less than 0.25, and Prediction Level (Pred) of 0.25 for 75% of the time (Conte, Dunsmore, & Shen, 1986:172-175).

In addition to running the COCOMO model in calibrated and uncalibrated mode, an attempt to improve accuracy by applying regression techniques was employed. The regression analysis also served to assist investigation of the model's bias in producing person month estimates.

The research questions to be answered included:

1. What is the uncalibrated accuracy of the COCOMO II model in the Post-Architecture mode when estimating efforts in the SMC SWDB for both the calibration and validation subsets?

2. What is the calibrated accuracy of the COCOMO II model in the Post-Architecture mode when estimating efforts in the SMC SWDB for both the calibration and validation subsets?

3. Are there any improvements in accuracy between the calibrated and uncalibrated settings of the COCOMO II model in the Post-Architecture mode?

4. In its current form, is the COCOMO II model useful for DOD cost analysts on software development projects?

The first two research questions can be answered from the tables below. The third

Table 23. Comparison of the Calibration Data Set Criteria

	CALIBRATION DATA SET RESULTS					
	Default Mode			*Calibrated Mode*		
APPLICATION TYPE	MMRE	RRMS	PRED(.25)	MMRE	RRMS	PRED(.25)
Mil Grd--C^2	0.3598	0.5216	0.4400	0.4037	0.4286	0.3000
Mil Grd--Signal Proc	0.4084	0.4999	0.3846	0.3890	0.4416	0.3692
Grd in Support of Space	0.5941	1.1072	0.2167	0.5885	0.7355	0.1167
Mil Mobile	0.6817	1.0699	0.0800	0.7030	0.8231	0.0800

Table 24. Comparison of the Validation Data Set Criteria

	VALIDATION DATA SET RESULTS					
	Default Mode			*Calibrated Mode*		
APPLICATION TYPE	MMRE	RRMS	PRED(.25)	MMRE	RRMS	PRED(.25)
Mil Grd--C^2	0.3933	0.4858	0.3000	0.3332	0.5318	0.4000
Mil Grd--Signal Proc	0.4507	0.6275	0.3333	0.3845	0.5343	0.4000
Grd in Support of Space	0.7077	1.1652	0.0667	0.6587	0.9498	0.2000
Mil Mobile	0.7930	0.9458	0.1000	0.6762	0.7381	0.0000

research question can be answered by comparing the results of the default and calibrated mode criterian for each data set. Surprisingly, we see that the calibration data set criteria worsened with calibration. This can be explained by the calibrated models increased accuracy for the higher effort projects, at the expense of a loss in accuracy for the lower effort projects. As anticipated, the criteria for the validation data set improved in all cases except Military Mobile, which, showed virtually no improvement and can be attributed to the fact that it was the only data set that was not homogeneous. The last research question is a little more difficult to answer. Although this study did not meet Conte's criteria, it's the belief of this researcher, that the COCOMO II model is useful in its current state based on the following two assumptions:

1. The Cost Analyst is experienced with COCOMO II and the project being estimated.
2. Homogeneous data is available to properly calibrate the model.

If these two assumptions cannot be met, then other methods are necessary to improve the accuracy. Other suggested methods of improving model accuracy when cost analyst experience and/or homogeneous data is not available include, requesting experienced assistance when estimating software effort and using regression techniques. The second part of this research was to apply regression to the default estimates.

In this researcher's opinion, the regression portion of this research proved highly successful and should be useful to DOD Cost Analysts in the field. The results of the analysis may be seen in the table below. Overall and after transformation, the regression equations improved the accuracy of the effort estimates significantly. What is not evident in the Pred(.25) column of the table is that there were still a number of estimates that were between 0.25 and 0.30 for each data set. For Pred(.25), if the estimate was greater than 0.2500, then it was not counted in the calculation.

Table 25. Accuracy Results of Final Regression Model

DEFAULT ACCURACY					Improved with Regression
	w/o Regression		*w/ Regression*		
APPLICATION TYPE	**MMRE**	**PRED(.25)**	**MMRE**	**PRED(.25)**	
Mil Grd--C^2	0.3671	0.4167	0.2059	0.7500	yes
Mil Grd--Signal Proc	0.4163	0.3750	0.3240	0.6250	yes
Grd in Support of Space	0.6168	0.2000	0.5140	0.3333	yes
Mil Mobile	0.7003	0.0833	0.7467	0.1667	no

Conclusions

Before the usefulness of the this research can be determined by an individual or organization for their specific needs, a review of the limitations and strengths/significant findings must be discussed.

The limitations of this research and accompanied discussion follow:

1. The Pred(.25) criterion doesn't take into account those estimates that fall close to 0.25. A Pred(.30) could have also been given, but what about those estimates that are slightly greater than 0.30? It was the view of this researcher that a cutoff point had to be maintained, but that the individual MRE data would be presented in the text for individual analysis.
2. The most significant weakness of this study was the data itself. Anomalies were identified, but the required support to either correct the data or eliminate it from the data set was not sufficient.
3. Although data sets were kept in tact for the regression runs, one data point was eliminated for both the Military Ground—Signal Processing and Ground in Support of Space data sets to facilitate an acceptable p-value when running the Shapiro-Wilk test. Nevertheless, it is not felt this was detrimental to the research and end results. The data point was chosen based on visual inspection of the residual plots, which indicated a severe outlier.
4. The equal variance test showed a calculated F value of less than one for the Ground in Support of Space data set. This indicates that model accuracy improves as effort estimates increase. This is counter to what one would expect, and after further investigation of the productivity rate, EMs, and size, and given the obvious conclusion (equation is better fit for higher effort), no solid conclusion could be determined for this reverse heteroschedastic megaphone.

5. Initially, there was concern that the lack of a language parameter was a source of weakness, however, after concluding the analysis phase of the research, it appears that the single greatest weakness of the model itself (not of this study), is a lack of risk simulation. Undoubtedly, the typical estimate in this research would have a probability of something less than 100%, since the model tends to underestimate as determined by the Wilkoxon-Signed Rank test and regression analysis. Including risk should bound the estimate with a high and low effort, and ideally, should always contain the final actual effort.

Given the above limitations, it's now beneficial to present the strengths/significant findings of the research so that the reader can determine adequacy of this research, based on his or her needs and requirements.

1. The COCOMO II model was very easy to use.
2. The equations within the model are not proprietary, and therefore, are visible to help the user have a greater understanding of how the model works and the effects of individual attributes upon the estimates. This model can easily be used in Microsoft Excel® by entering in the equations and the EM table, which enhances the user's ability to make multiple calculations for calibration purposes, including risk analysis (would need to use an Microsoft Excel® based risk program), and print options.
3. It's better to use a homogeneous data set when calibrating the model than a heterogeneous data set. Based on Pred(.25), accuracy does improve with calibration in all cases except for the one heterogeneous data set used in this study (Military Mobile); based solely on RRMS and MMRE, there was mixed improvement among the data sets.
4. Data quality and Cost Analyst experience with the platform in question are undoubtedly the most important factors to achieving the most accurate

estimates from the model. Cost Analyst experience with the parametric software cost model is also critical.

5. When data is weak and doesn't contain the necessary attribute information (to determine EAF values), it appears that setting the EM to nominal is a viable alternative to trying to get perfect information. This is based on the results of the Military Ground—Signal Processing data set.
6. Regression techniques applied to the default model estimates will improve estimating accuracy.
7. The USER 1 and USER 2 parameters available in the COCOMO II model appear beneficial for the experienced analyst.

The emphasis for presentation of the limitations, the strengths, and the analysis of this study was to be objective, provide useful data, and present all the limiting factors, so that the reader may use sound judgment in determining the validity of this research to his or her work environment.

Recommendations

COCOMO II appears to be a viable software estimating tool for the cost analyst, and for DOD. It's highly recommended that this model be used by all Cost Analysts in DOD as either a primary or secondary software cost estimating model, for the following reasons:

1. It is manageable by the user and has visibility of equations and model functionality;
2. It is of no cost to the government, except for any funding they may provide for COCOMO research;

3. It is simple to use and not overloaded with unnecessary parameters (in other words, it follows the Principle of Parsimony), which can equate to more time using the model and gathering data;
4. Based on previous studies (see Table 1), other software cost estimating models do not appear to be any more accurate than the COCOMO II model.

Quality, robust data and experienced model users will have a great impact on the accuracy of the model. However, when this is not possible, the use of regression techniques can be used to improve the overall accuracy of the estimates.

It's recommended that future research efforts use what has been accomplished here to alleviate the up front work an allow for a greater focus of new research on completing one of the two following topics.

1. Research the data for accuracy and anomalies. Determine actual outliers that may still exist, and either correct or eliminate. Run COCOMO then to produce new estimates with the better data. The downfall of this research is that it could be expensive. Personal contact and communication with MCR, contractors, and SMC will be necessary to improve the data. All AFIT studies have been hindered by a lack of sound, robust data, and this hypothesized limitation needs to be validated once and for all.
2. Since the COCOMO II equations are available for use, it would be advantageous to develop a model in Microsoft Excel® that incorporated the use of Monte Carlo Simulation for DOD. Probability Distribution Functions would have to be determined for the EAFs and SLOC, which would result in a risk based estimate.
3. Calibrate to ESC database to determine if individual contractors can result in better accuracy.

Contrary to previous AFIT theses, it doesn't appear evident from this research that the model performed any better with certain applications. In fact, it seems that the model accuracy was linked to whether the data set was homogeneous. The future research described above could help to prove or disprove whether the model is applicable to different applications equally well. If this is so, then it would be to the benefit of the DOD to use COCOMO II as a primary (or secondary) software cost model versus having the expense of a commercial model.

Appendix A. Acronyms and Glossary of Terms

Analogy - A method of comparing like systems and applying a factor to derive a new estimate.

ANOVA Table - ANalysis Of VAriance Table

AFCAA - Air Force Cost Analysis Agency.

AFIT - Air Force Institute of Technology.

Application - The type of software package, i.e. Military Ground, MIS, or Avionics software are examples of applications or platforms.

C^2 - Command & Control.

CAIV - Cost As an Independent Variable.

CER - Cost Estimating Relationship.

CSCI - Computer Software Configuration Item.

CMM - Capability Maturity Model; developed by Software Engineering Institute.

COCOMO - COnstructive COst MOdel.

COTS - Commercial-off-the-shelf.

DAF - Department of the Air Force.

Effort - For software, this equates to Person Months (PM) required to complete a task, phase, or project.

ESC - Electronic Systems Center.

Expert Opinion - The use of those knowledgeable of a system to assist in deriving analogies and relationships for the system being estimated.

Heteroschedastic - Unequal variance of residual values. For this study, specific attention was given to residuals that displayed a megaphone style of heteroschedasticity, which is simply increasing variance as the independent variables increased in value.

Homoschedastic - Equal variance of residual values.

IFPUG - International Function Point Users Group.

IPT - Integrated Product Team; a team of functional experts (logistics, cost analyst, program manager, engineer, contractor) brought together to determine the best method of procuring, modifying, or addressing issues in a SPO.

Metric - A snapshot measure used to quantify system progress.

MIS - Management Information System—business software, i.e. accounts payable software.

MRE - Magnitude of Relative Error = |(Estimate - Actual)/Actual|
The degree of estimating error in an individual estimate.

MMRE - Mean Magnitude of Relative Error = Sum (MRE/n)
The average degree of estimating error in a data set.

Nonparametric - "A statistical method is nonparametric if it satisfies at least one of the following criteria.

1. The method may be used on data with a nominal scale of measurement.
2. The method may be used on data with an ordinal scale of measurement.
3. The method may be used on data with an interval or ratio scale of measurement, where the distribution function of the random variable producing the data is either unspecified or specified except for an infinite number of unknown parameters" (Conover, 1980:92).

Object Oriented Design - A methodology of writing code in block form, which consists of only the code necessary to perform a specific routine, function etc.

Object Points - Similar to Function points, however, it is a count of rule sets, third generation languages, screen definitions, and user reports (Ferens, 1997).

Parametric - Uses cost estimating relationships, such as regression, or algorithms to make estimates.

PDF - Probability Distribution Function.

Platform - same as application, defined above.

Prediction Level - The number of records (k) that fall within a specified limit (given in percentage) divided by the total number of records (n).
Pred (.25) = k/n The percentage of estimates within 25% of the actual results.

Regression - A means of developing a functional relationship between at least one independent variable, and one dependent variable.

RMS - Root Mean Square = [1/n Sum (Estimate - Actual)^2]^0.5
The model's ability to accurately forecast the individual actual effort.

RRMS - Relative Root Mean Square = RMS/[Sum (Actual)/n]
The model's ability to accurately forecast the average actual effort.

Software Breakage - Percentage of code thrown away due to requirements volatility.

Software process - a set of activities, methods, practices, and transformations that people employ to develop and maintain software and the associated products (CMU, et al., 1995: 8).

SMC SWDB - Space and Missile Systems Center Software Database.

SP - Specific Avionics.

SPO - Systems Program Office.

TQM - Total Quality Management.

Appendix B. SMC SWDB Data Sets

Military Ground—Command and Control, SMC SWDB Data Set

NUMBER	KEY FID	HRS/ PM	TOTSZ _NORM	EFFRT _NORM	QA_ LEV	APPL _DIFF	REUSE _REQM	PERS _CAP	PERS _EXP	TEAM_ LANG_	DEV_ METH _E	AUTO _TOOLS	VOLAT
1	0007	-1.00	46400	120	Low	High	Nominal			Nominal		Nominal	Low
2	0009	-1.00	128200	517	Low	High	Nominal			Nominal		Nominal	Low
3	0050	-1.00	144000	684	Very Low	High	Nominal			Nominal		Low	Low
4	0120	-1.00	25842	95									
5	0124	-1.00	23881	139									
6	0141	-1.00	162039	322									
7	0145	-1.00	18560	101									
8	0150	-1.00	21681	100									
9	0152	-1.00	69772	286									
10	0155	-1.00	8398	74									
11	2510	153.00	43437	172	Nominal	High	Nominal	Nominal	High	High	High	Nominal	Low
12	2517	160.00	90000	167	Low	High	Nominal	High	High	Very High	High	Low	Nominal

<u>Note</u>: All projects from this Application data set were used.

Military Ground—Signal Processing, SMC SWDB Data Set

NUMBER	KEY FLD	HRS/PM	TOTSZ_NORM	EFFRT_NORM	QA_LEV	APPL DIFF	REUSE_REQM	PERS_CAP	PERS_EXP	TEAM_LANG_	DEV_METH_E	AUTO_TOOLS	VOLAT
1	0054	-1.00	45700	127		Very High	Very High			Low		Very Low	
2	0126	-1.00	47965	165									
3	0127	-1.00	16016	13									
4	0130	-1.00	71851	738									
5	0131	-1.00	29147	192									
6	0132	-1.00	46595	278									
7	0133	-1.00	123710	645									
8	0134	-1.00	44527	228									
9	0135	-1.00	23787	264									
10	0136	-1.00	12121	154									
11	0137	-1.00	60233	274									
12	0138	-1.00	14389	190									
13	0140	-1.00	70020	6									
14	0142	-1.00	28782	348									
15	0143	-1.00	23703	86									
16	0144	-1.00	29802	145									
17	0147	-1.00	31720	192									
18	0153	-1.00	11534	149									
19	0154	-1.00	8965	109									

Note: Projects 1, 3, and 13 were eliminated from the data base due to inconsistencies between the productivity rates and EMs.

See page 60 for details.

Ground in Support of Space, SMC SWDB Data Set

NUMBER	KEY FLD	HRS/ PM	TOTSZ _NORM	EFFRT _NORM	QA_ LEV	APPL DIFF	REUSE _REQM	PERS _CAP	PERS _EXP	TEAM_ LANG_	DEV_ METH _E	AUTO _TOOLS	VOLAT
1	0038	-1.00	6000	61	Nominal	Extra High	High			Very Low		Nominal	Low
2	0074	-1.00	11700	80	Nominal	High	Nominal			Low	Low	Nominal	High
3	0075	-1.00	116800	912	Nominal	Very High	Nominal			Low	Low	Nominal	High
4	0076	-1.00	14000	115	Nominal	Nominal	Nominal			Nominal	Nominal	Nominal	High
5	0077	-1.00	56200	523	Nominal	Very High	Nominal			Low	Low	Nominal	High
6	0078	-1.00	48300	478	Nominal	Nominal	Nominal			Low	Low	Nominal	High
7	0079	-1.00	50300	432	Nominal	Nominal	Nominal			Low	Low	Nominal	High
8	0080	-1.00	69450	296	Nominal	Nominal	Nominal			Low	Low	Nominal	High
9	0081	-1.00	22900	164	Nominal	Nominal	Nominal			Low	Low	Nominal	High
10	0082	-1.00	16300	140	Nominal	Nominal	Nominal			Low	Low	Nominal	High
11	0083	-1.00	6800	57	Nominal	Low	Nominal			Low	Low	Nominal	High
12	0093	-1.00	250000	401									
13	0119	-1.00	278488	787									
14	0329	-1.00	34650	57	Low	Nominal	High			High	Nominal	Nominal	Nominal
15	0331	-1.00	7000	18	Nominal	High	Nominal			High	High	Nominal	Nominal

Note: All projects from this Application data set were used.

Military Mobile, SMC SWDB Data Set

NUMBER	KEY FLD	HRS/ PM	TOT SZ_ NORM	EFFRT _NORM	QA_ LEV	APPL DIFF	REUSE _REQM	PERS _CAP	PERS _EXP	TEAM_ LANG_	DEV_ METH _E	AUTO _TOOLS	VOLAT
1	0034	-1.00	17350	83						Nominal			
2	0303	-1.00	30000	237	Nominal	Extra High	Nominal			Nominal	Low	Nominal	Nominal
3	0347	-1.00	2311	39	Nominal	Very High	High			Nominal	Very High	Nominal	High
4	0348	-1.00	18052	396	Nominal	Very High	High			Very Low	Low	Nominal	High
5	0349	-1.00	3268	56						High	Nominal	Nominal	
6	2456	-1.00	88633	221			Nominal	Nominal	Very High				
7	2502	150.00	26239	633	Nominal	Nominal	Nominal	Nominal	Nominal	Low	Low	Nominal	Nominal
8	2503	150.00	32464	783	Nominal	Nominal	Nominal	Nominal	Nominal	Low	Low	Nominal	Nominal
9	2505	150.00	7448	180	Nominal	Nominal	Nominal	Nominal	Nominal	Low	Low	Nominal	Nominal
10	2506	150.00	6317	152	Nominal	Nominal	Nominal	Nominal	Nominal	Low	Low	Nominal	Nominal
11	2507	150.00	26814	647	Nominal	Nominal	Nominal	Nominal	Nominal	Low	Low	Nominal	Nominal
12	2508	150.00	58789	1418	Nominal	Nominal	Very High	Nominal	Nominal	Low	Low	Nominal	Nominal

Note: All projects from this Application data set were used.

Military Ground—Command and Control, EAF and EM Values

	Keyfields (Records)											
	1	2	3	4	5	6	7	8	9	10	11	12
Exponent Adjustment Factors (EAFs)												
Reliability (RELY)	0.88	0.88	0.75	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.88
Data (DATA)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Documentation (DOCU)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Complexity (CPLX)	1.15	1.15	1.15	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.15	1.15
Reusability (RUSE)	0.91	0.91	0.91	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.91	0.91
Time (TIME)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Storage (STOR)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Platform Volatility (PVOL)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Analyst Capability (ACAP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.87
Applications Experience (AEXP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.89	0.89
Programmer Capability (PCAP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.88
Platform Experience (PEXP)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Language Team Experience (LTEX)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Personnel Continuity (PCON)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Use of Software Tools (TOOL)	1.00	1.00	1.12	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.12
Multisite Development (SITE)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Schedule (SCED)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Devpmnt Sys Volat. User 1(USR 1)	0.87	0.87	0.87	0.87	0.87	0.87	0.87	0.87	0.87	0.87	0.87	0.87
User 2 (USR2)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
EM =	**.8012**	**.8012**	**.7648**	**.8700**	**.8700**	**.8700**	**.8700**	**.8700**	**.8700**	**.8700**	**.8103**	**.6114**

Military Ground—Signal Processing, EAF and EM Values

	Keyfields (Records)															
	2	4	5	6	7	8	9	10	11	12	14	15	16	17	18	19
Exponent Adjustment Factors (EAFs)																
Reliability (RELY)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Data (DATA)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Documentation (DOCU)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Complexity (CPLX)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Reusability (RUSE)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Time (TIME)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Storage (STOR)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Platform Volatility (PVOL)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Analyst Capability (ACAP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Applications Exp (AEXP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Programmer Capability (PCAP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Platform Exp (PEXP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Lang Team Exp (LTEX)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Personnel Continuity (PCON)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Use of Software Tools (TOOL)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Multisite Development (SITE)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Schedule (SCED)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Dev Sys Volat. User 1(USR 1)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
User 2 (USR2)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
EM =	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

Note: Projects 1, 3, and 13 were eliminated from the data base due to inconsistencies between the productivity rates and EMs.

See page 60 for details.

Ground in Support of Space, EAF and EM Values

	Keyfields (Records)														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Exponent Adjust Factors (EAFs)															
Reliability (RELY)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.88	1.00
Data (DATA)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Document (DOCU)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Complexity (CPLX)	1.66	1.15	1.30	1.00	1.30	1.00	1.00	1.00	1.00	1.00	0.88	1.00	1.00	1.00	1.15
Reusability (RUSE)	1.00	0.91	0.91	0.91	0.91	0.91	0.91	0.91	0.91	0.91	0.91	1.00	1.00	1.00	0.91
Time (TIME)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Storage (STOR)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Platfm Volat (PVOL)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Analyst Cap (ACAP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Applic Exp (AEXP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Programr Cap (PCAP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Platform Exp (PEXP)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Lang Team Exp (LTEX)	1.22	1.10	1.10	1.00	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.00	1.00	1.00	1.00
Personnel Continuity (PCON)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Use of s/w Tls (TOOL)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Multisite Devp (SITE)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Schedule (SCED)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Devp Sys Volat. User 1(USR 1)	0.87	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.87	0.87
User 2 (USR2)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
EM =	**1.762**	**1.151**	**1.301**	**0.910**	**1.301**	**1.001**	**1.001**	**1.001**	**1.001**	**1.001**	**0.881**	**1.000**	**1.000**	**0.766**	**0.911**

Military Mobile, EAF and EM Values

	Keyfields (Records)											
	1	2	3	4	5	6	7	8	9	10	11	12
Exponent Adjustment Factors (EAFs)												
Reliability (RELY)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Data (DATA)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Documentation (DOCU)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Complexity (CPLX)	1.00	1.66	1.30	1.30	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Reusability (RUSE)	1.00	0.91	1.00	1.00	1.00	0.91	0.91	0.91	0.91	0.91	0.91	1.29
Time (TIME)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Storage (STOR)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Platform Volatility (PVOL)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Analyst Capability (ACAP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Applications Experience (AEXP)	1.00	1.00	1.00	1.00	1.00	0.81	1.00	1.00	1.00	1.00	1.00	1.00
Programmer Capability (PCAP)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Platform Experience (PEXP)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Language Team Experience (LTEX)	1.00	1.10	1.00	1.22	1.00	1.00	1.10	1.10	1.10	1.10	1.10	1.10
Personnel Continuity (PCON)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Use of Software Tools (TOOL)	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Multisite Development (SITE)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Schedule (SCED)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
Devpmnt Sys Volat, User 1(USR 1)	1.00	0.87	1.00	1.00	1.00	1.00	0.87	0.87	0.87	0.87	0.87	0.87
User 2 (USR2)	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*	*1.00*
EM =	**1.000**	**1.446**	**1.300**	**1.586**	**1.000**	**0.737**	**0.871**	**0.871**	**0.871**	**0.871**	**0.871**	**1.235**

Military Ground—Command and Control, Productivity/EM Comparison

PROJ	SWDB NOR EFFORT	S/W REQM ADDED	FINAL EFFORT	SIZE (KSLOC)	PRODUCTIVITY	EM
1	120	6.60	126.60	46.40	0.367	0.8012
2	517	28.44	545.44	128.20	0.235	0.8012
3	684	37.62	721.62	144.00	0.200	0.7648
4	95	5.23	100.23	25.84	0.258	0.8700
5	139	7.65	146.65	23.88	0.163	0.8700
6	322	17.71	339.71	162.04	0.477	0.8700
7	101	5.56	106.56	18.56	0.174	0.8700
8	100	5.50	105.50	21.68	0.206	0.8700
9	286	15.73	301.73	69.77	0.231	0.8700
10	74	4.07	78.07	8.40	0.108	0.8700
11	172	9.46	181.46	43.44	0.239	0.8103
12	167	9.19	176.19	90.00	0.511	0.6114

Military Ground—Signal Processing, Productivity/EM Comparison

PROJ	SWDB NOR EFFORT	S/W REQM ADDED	FINAL EFFORT	SIZE (KSLOC)	PRODUCTIVITY	EM
1	127	6.99	133.99	45.70	0.341	**2.2874**
2	165	9.08	174.08	47.97	0.276	1.0000
3	13	0.72	13.72	16.02	**1.168**	1.0000
4	738	40.59	778.59	71.85	0.092	1.0000
5	192	10.56	202.56	29.15	0.144	1.0000
6	278	15.29	293.29	46.60	0.159	1.0000
7	645	35.48	680.48	123.71	0.182	1.0000
8	228	12.54	240.54	44.53	0.185	1.0000
9	264	14.52	278.52	23.79	0.085	1.0000
10	154	8.47	162.47	12.12	0.075	1.0000
11	274	15.07	289.07	60.23	0.208	1.0000
12	190	10.45	200.45	14.39	0.072	1.0000
13	6	0.33	6.33	70.02	**11.062**	1.0000
14	348	19.14	367.14	28.78	0.078	1.0000
15	86	4.73	90.73	23.70	0.261	1.0000
16	145	7.98	152.98	29.80	0.195	1.0000
17	192	10.56	202.56	31.72	0.157	1.0000
18	149	8.20	157.20	11.53	0.073	1.0000
19	109	6.00	115.00	8.97	0.078	1.0000

<u>Note</u>: Projects 1, 3, and 13 were eliminated from the data base due to inconsistencies between the productivity rates and EMs.

See page 60 for details.

Ground in Support of Space, Productivity/EM Comparison

PROJ	SWDB NOR EFFORT	S/W REQM ADDED	FINAL EFFORT	SIZE (KSLOC)	PRODUCTIVITY	EM
1	61	3.36	64.36	6.00	0.093	1.7619
2	80	4.40	84.40	11.70	0.139	1.1512
3	912	50.16	962.16	116.80	0.121	1.3013
4	115	6.33	121.33	14.00	0.115	0.9100
5	523	28.77	551.77	56.20	0.102	1.3013
6	478	26.29	504.29	48.30	0.096	1.0010
7	432	23.76	455.76	50.30	0.110	1.0010
8	296	16.28	312.28	69.45	0.222	1.0010
9	164	9.02	173.02	22.90	0.132	1.0010
10	140	7.70	147.70	16.30	0.110	1.0010
11	57	3.14	60.14	6.80	0.113	0.8809
12	401	22.06	423.06	250.00	0.591	1.0000
13	787	43.29	830.29	278.49	0.335	1.0000
14	57	3.14	60.14	34.65	0.576	0.7656
15	18	0.99	18.99	7.00	0.369	0.9105

Military Mobile, Productivity/EM Comparison

PROJ	SWDB NOR EFFORT	S/W REQM ADDED	FINAL EFFORT	SIZE (KSLOC)	PRODUCTIVITY	EM
1	83	4.57	87.57	17.35	0.198	1.0000
2	237	13.04	250.04	30.00	0.120	1.4456
3	39	2.15	41.15	2.31	0.056	1.3000
4	396	21.78	417.78	18.05	0.043	1.5860
5	56	3.08	59.08	3.27	0.055	1.0000
6	221	12.16	233.16	88.63	0.380	0.7371
7	633	34.82	667.82	26.24	0.039	0.8709
8	783	43.07	826.07	32.46	0.039	0.8709
9	180	9.90	189.90	7.45	0.039	0.8709
10	152	8.36	160.36	6.32	0.039	0.8709
11	647	35.59	682.59	26.81	0.039	0.8709
12	1418	77.99	1495.99	58.79	0.039	1.2345

Appendix C. Project Selection for Calibration and Validation

Military Ground—Command and Control, Calibration & Validation Subset Generation

	PROJECT NUMBER											
RUN	1	2	3	4	5	6	7	8	9	10	11	12
1		X	X	X	X	X		X	X	X	X	X
2	X	X	X	X	X	X	X	X		X	X	
3	X	X	X	X	X		X		X	X	X	X
4	X	X	X	X	X	X	X	X	X	X		
5	X		X	X	X	X		X	X	X	X	X

Note: Calibration is denoted with an 'X', and validation is left blank.

Military Ground—Signal Processing, Calibration & Validation Subset Generation

	PROJECT NUMBER															
RUN	2	4	5	6	7	8	9	10	11	12	14	15	16	17	18	19
1	X	X	X	X	X	X	X	X	X		X			X	X	X
2			X	X	X	X	X	X	X	X		X	X	X	X	X
3	X	X	X	X	X	X	X		X	X		X		X	X	X
4	X	X		X	X	X	X	X	X	X		X	X		X	X
5	X	X		X	X		X	X	X	X		X	X	X	X	X

Note: Calibration is denoted with an 'X', and validation is left blank.

Ground in Support of Space, Calibration & Validation Subset Generation

	PROJECT NUMBER														
RUN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	X	X	X	X	X	X	X	X	X				X	X	X
2	X	X			X	X		X	X	X	X	X	X	X	X
3		X	X	X		X	X	X		X	X	X	X	X	X
4		X	X	X	X	X	X	X	X	X	X	X			X
5	X	X	X		X		X	X	X	X	X	X	X		X

Note: Calibration is denoted with an 'X', and validation is left blank.

Military Mobile, Calibration & Validation Subset Generation

	PROJECT NUMBER											
RUN	1	2	3	4	5	6	7	8	9	10	11	12
1	X	X	X	X	X	X	X		X	X	X	
2	X	X	X	X		X	X	X	X		X	X
3	X	X	X	X	X	X		X	X		X	X
4	X	X		X	X	X	X	X		X	X	X
5		X	X	X	X	X	X	X		X	X	X

Appendix C1. Coefficient Calibration

Military Ground—Command & Control Coefficient Calibration, Run 1

Project	SIZE	PM	EM	Q	PM*Q	Q2
2	128.20	545.435	0.8012	215.8420467	117727.8	46587.79
3	144.00	721.620	0.7648	235.5752848	169995.8	55495.71
4	25.84	100.225	0.8700	36.97696579	3706.016	1367.296
5	23.88	146.645	0.8700	33.76088286	4950.865	1139.797
6	162.04	339.710	0.8700	307.0510584	104308.3	94280.35
8	21.68	105.500	0.8700	30.20081732	3186.186	912.0894
9	69.77	301.730	0.8700	116.2207421	35067.28	13507.26
10	8.40	78.070	0.8700	10.117985	789.9111	102.3736
11	43.44	181.460	0.8103	62.67594166	11373.18	3928.274
12	90.00	176.185	0.6114	109.5450417	19300.19	12000.12
				SUM	470405.6	229321.1

A = 2.0513

Military Ground—Command & Control Coefficient Calibration, Run 2

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	46	126.60	0.8012	66.87078001	8465.841	4471.701
2	128	545.44	0.8012	223.3013078	121796.3	49863.47
3	144	721.62	0.7648	243.9148615	176013.8	59494.46
4	26	100.23	0.8700	37.82836356	3791.348	1430.985
5	24	146.65	0.8700	34.51915552	5062.062	1191.572
6	162	339.71	0.8700	318.1837058	108090.2	101240.9
7	19	106.56	0.8700	25.24583004	2690.196	637.3519
8	22	105.50	0.8700	30.85824704	3255.545	952.2314
10	8	78.07	0.8700	10.26983018	801.7656	105.4694
11	43	181.46	0.8103	64.35256863	11677.42	4141.253
				SUM	441644.6	223529.4

A = 1.9758

Military Ground—Command & Control Coefficient Calibration, Run 3

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	46	126.60	0.8012	66.87078001	8465.841	4471.701
2	128	545.44	0.8012	223.3013078	121796.3	49863.47
3	144	721.62	0.7648	243.9148615	176013.8	59494.46
4	26	100.23	0.8700	37.82836356	3791.348	1430.985
5	24	146.65	0.8700	34.51915552	5062.062	1191.572
7	19	106.56	0.8700	25.24583004	2690.196	637.3519
9	70	301.73	0.8700	119.7262589	36125	14334.38
10	8	78.07	0.8700	10.26983018	801.7656	105.4694
11	43	181.46	0.8103	64.35256863	11677.42	4141.253
12	90	176.19	0.6114	113.0504831	19917.8	12780.41
				SUM	386341.6	148451.1

A = 2.6025

Military Ground—Command & Control Coefficient Calibration, Run 4

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	46.40	126.60	0.8012	66.87078001	8465.841	4471.701
2	128.20	545.44	0.8012	223.3013078	121796.3	49863.47
3	144.00	721.62	0.7648	243.9148615	176013.8	59494.46
4	25.84	100.23	0.8700	37.82836356	3791.348	1430.985
5	23.88	146.65	0.8700	34.51915552	5062.062	1191.572
6	162.04	339.71	0.8700	318.1837058	108090.2	101240.9
7	18.56	106.56	0.8700	25.24583004	2690.196	637.3519
8	21.68	105.50	0.8700	30.85824704	3255.545	952.2314
9	69.77	301.73	0.8700	119.7262589	36125	14334.38
10	8.40	78.07	0.8700	10.26983018	801.7656	105.4694
				SUM	466092.1	233722.5

A = 1.9942

Military Ground—Command & Control Coefficient Calibration, Run 5

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	46.40	126.60	0.8012	66.87078001	8465.841	4471.701
3	144.00	721.62	0.7648	243.9148615	176013.8	59494.46
4	25.84	100.23	0.8700	37.82836356	3791.348	1430.985
5	23.88	146.65	0.8700	34.51915552	5062.062	1191.572
6	162.04	339.71	0.8700	318.1837058	108090.2	101240.9
8	21.68	105.50	0.8700	30.85824704	3255.545	952.2314
9	69.77	301.73	0.8700	119.7262589	36125	14334.38
10	8.40	78.07	0.8700	10.26983018	801.7656	105.4694
11	43.44	181.46	0.8103	64.35256863	11677.42	4141.253
12	90.00	176.19	0.6114	113.0504831	19917.8	12780.41
				SUM	373200.8	200143.3

A = 1.8647

Military Ground—Signal Processing Coefficient Calibration, Run 1

Project	SIZE	PM	EM	Q	PM*Q	Q2
2	47.97	174.08	1.0000	86.71733	15095.32	7519.895
4	71.85	778.59	1.0000	138.187	107591	19095.63
5	29.15	202.56	1.0000	48.82888	9890.777	2384.259
6	46.60	293.29	1.0000	83.8678	24597.59	7033.807
7	123.71	680.48	1.0000	258.5489	175936.1	66847.55
8	44.53	240.54	1.0000	79.59079	19144.77	6334.694
9	23.79	278.52	1.0000	38.62955	10759.1	1492.242
10	12.12	162.47	1.0000	17.75497	2884.649	315.2388
11	60.23	289.07	1.0000	112.7585	32595.09	12714.47
14	28.78	367.14	1.0000	48.12453	17668.44	2315.97
17	31.72	202.56	1.0000	53.83158	10904.13	2897.839
18	11.53	157.195	1.0000	16.76729	2635.734	281.142
19	8.97	114.995	1.0000	12.5398	1442.014	157.2465
				SUM	431144.7	129390

A = 3.3321

Military Ground—Signal Processing Coefficient Calibration, Run 2

Project	SIZE	PM	EM	Q	PM*Q	Q2
5	29.147	202.56	1.0000	49.99527	10127.04	2499.527
6	46.595	293.29	1.0000	86.15364	25268	7422.449
7	123.71	680.475	1.0000	267.4173	181970.8	71512.04
8	44.527	240.54	1.0000	81.73408	19660.32	6680.46
9	23.787	278.52	1.0000	39.49608	11000.45	1559.941
10	12.121	162.47	1.0000	18.06777	2935.471	326.4444
11	60.233	289.07	1.0000	116.0401	33543.71	13465.3
12	14.389	200.45	1.0000	21.63761	4337.259	468.1862
15	23.703	90.73	1.0000	38.47231	3490.592	1480.118
16	29.802	152.975	1.0000	50.09622	7663.469	2509.631
17	31.72	202.56	1.0000	55.15012	11171.21	3041.536
18	11.534	157.195	1.0000	17.05677	2681.239	290.9333
19	8.965	114.995	1.0000	12.73381	1464.324	162.1499
				SUM	315313.9	111418.7

A = 2.8300

Military Ground—Signal Processing Coefficient Calibration, Run 3

Project	SIZE	PM	EM	Q	PM*Q	Q2
2	47.965	174.075	1.0000	86.71733	15095.32	7519.895
4	71.851	778.59	1.0000	138.187	107591	19095.63
5	29.147	202.56	1.0000	48.82888	9890.777	2384.259
6	46.595	293.29	1.0000	83.8678	24597.59	7033.807
7	123.71	680.475	1.0000	258.5489	175936.1	66847.55
8	44.527	240.54	1.0000	79.59079	19144.77	6334.694
9	23.787	278.52	1.0000	38.62955	10759.1	1492.242
11	60.233	289.07	1.0000	112.7585	32595.09	12714.47
12	14.389	200.45	1.0000	21.63761	4337.259	468.1862
15	23.703	90.73	1.0000	38.47231	3490.592	1480.118
17	31.72	202.56	1.0000	53.83158	10904.13	2897.839
18	11.534	157.195	1.0000	16.76729	2635.734	281.142
19	8.965	114.995	1.0000	12.5398	1442.014	157.2465
				SUM	418419.4	128707.1

A = 3.2509

Military Ground—Signal Processing Coefficient Calibration, Run 4

Project	SIZE	PM	EM	Q	PM*Q	Q2
2	47.97	174.08	1.0000	86.71733	15095.32	7519.895
4	71.85	778.59	1.0000	138.187	107591	19095.63
6	46.60	293.29	1.0000	83.8678	24597.59	7033.807
7	123.71	680.48	1.0000	258.5489	175936.1	66847.55
8	44.53	240.54	1.0000	79.59079	19144.77	6334.694
9	23.79	278.52	1.0000	38.62955	10759.1	1492.242
10	12.12	162.47	1.0000	17.75497	2884.649	315.2388
11	60.23	289.07	1.0000	112.7585	32595.09	12714.47
12	14.39	200.45	1.0000	21.63761	4337.259	468.1862
15	23.70	90.73	1.0000	38.47231	3490.592	1480.118
16	29.80	152.975	1.0000	50.09622	7663.469	2509.631
18	11.53	157.195	1.0000	16.76729	2635.734	281.142
19	8.97	114.995	1.0000	12.5398	1442.014	157.2465
				SUM	408172.6	126249.9

A = 3.2331

Military Ground—Signal Processing Coefficient Calibration, Run 5

Project	SIZE	PM	EM	Q	PM*Q	Q2
2	47.97	174.08	1.0000	86.71733	15095.32	7519.895
4	71.85	778.59	1.0000	138.187	107591	19095.63
6	46.60	293.29	1.0000	83.8678	24597.59	7033.807
7	123.71	680.48	1.0000	258.5489	175936.1	66847.55
9	23.79	278.52	1.0000	38.62955	10759.1	1492.242
10	12.12	162.47	1.0000	17.75497	2884.649	315.2388
11	60.23	289.07	1.0000	112.7585	32595.09	12714.47
12	14.39	200.45	1.0000	21.63761	4337.259	468.1862
15	23.70	90.73	1.0000	38.47231	3490.592	1480.118
16	29.80	152.975	1.0000	50.09622	7663.469	2509.631
17	31.72	202.56	1.0000	53.83158	10904.13	2897.839
18	11.53	157.195	1.0000	16.76729	2635.734	281.142
19	8.97	114.995	1.0000	12.5398	1442.014	157.2465
				SUM	399932	122813

A = 3.2564

Ground in Support of Space Coefficient Calibration, Run 1

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	6.00	64.36	1.7619	13.90579	894.9069	193.3709
2	11.70	84.40	1.1512	19.62231	1656.123	385.0352
3	116.80	962.16	1.3013	314.8756	302960.7	99146.65
4	14.00	121.33	0.9100	19.07774	2314.607	363.9603
5	56.20	551.77	1.3013	135.4638	74744.21	18350.45
6	48.30	504.29	1.0010	87.50344	44127.11	7656.852
7	50.30	455.76	1.0010	91.69422	41790.56	8407.83
8	69.45	312.28	1.0010	133.0094	41536.16	17691.49
9	22.90	173.02	1.0010	37.01045	6403.548	1369.774
13	278.49	830.29	1.0000	658.9643	547128.2	434233.9
14	34.65	60.14	0.7656	45.63308	2744.145	2082.378
15	7.00	18.99	0.9105	8.583344	162.9977	73.6738
				SUM	1066463	589955.4

A= 1.8077

Ground in Support of Space Coefficient Calibration, Run 2

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	6.00	64.36	1.7619	13.90579	894.9069	193.3709
2	11.70	84.40	1.1512	19.62231	1656.123	385.0352
5	56.20	551.77	1.3013	135.4638	74744.21	18350.45
6	48.30	504.29	1.0010	87.50344	44127.11	7656.852
8	69.45	312.28	1.0010	133.0094	41536.16	17691.49
9	22.90	173.02	1.0010	37.01045	6403.548	1369.774
10	16.30	147.70	1.0010	25.00843	3693.745	625.4214
11	6.80	60.14	0.8809	8.031553	482.9774	64.50584
12	250.00	423.06	1.0000	581.8685	246162.4	338570.9
13	278.49	830.29	1.0000	658.9643	547128.2	434233.9
14	34.65	60.14	0.7656	45.63308	2744.145	2082.378
15	7.00	18.99	0.9105	8.583344	162.9977	73.6738
				SUM	969736.5	821297.8

A= 1.1807

Ground in Support of Space Coefficient Calibration, Run 3

Project	SIZE	PM	EM	Q	PM*Q	Q2
2	11.7	84.4	1.1512	19.62231	1656.123	385.0352
3	116.8	962.16	1.3013	314.8756	302960.7	99146.65
4	14	121.325	0.9100	19.07774	2314.607	363.9603
6	48.3	504.29	1.0010	87.50344	44127.11	7656.852
7	50.3	455.76	1.0010	91.69422	41790.56	8407.83
8	69.45	312.28	1.0010	133.0094	41536.16	17691.49
10	16.3	147.7	1.0010	25.00843	3693.745	625.4214
11	6.8	60.135	0.8809	8.031553	482.9774	64.50584
12	250	423.055	1.0000	581.8685	246162.4	338570.9
13	278.488	830.285	1.0000	658.9643	547128.2	434233.9
14	34.65	60.135	0.7656	45.63308	2744.145	2082.378
15	7	18.99	0.9105	8.583344	162.9977	73.6738
				SUM	1234760	909302.7

A= 1.3579

Ground in Support of Space Coefficient Calibration, Run 4

Project	SIZE	PM	EM	Q	PM*Q	Q2
2	11.7	84.4	1.1512	19.62231	1656.123	385.0352
3	116.8	962.16	1.3013	314.8756	302960.7	99146.65
4	14	121.325	0.9100	19.07774	2314.607	363.9603
5	56.2	551.765	1.3013	135.4638	74744.21	18350.45
6	48.3	504.29	1.0010	87.50344	44127.11	7656.852
7	50.3	455.76	1.0010	91.69422	41790.56	8407.83
8	69.45	312.28	1.0010	133.0094	41536.16	17691.49
9	22.9	173.02	1.0010	37.01045	6403.548	1369.774
10	16.3	147.7	1.0010	25.00843	3693.745	625.4214
11	6.8	60.135	0.8809	8.031553	482.9774	64.50584
12	250	423.055	1.0000	581.8685	246162.4	338570.9
15	7	18.99	0.9105	8.583344	162.9977	73.6738
				SUM	766035.1	492706.6

A= 1.5547

Ground in Support of Space Coefficient Calibration, Run 5

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	6	64.355	1.7619	13.90579	894.9069	193.3709
2	11.7	84.4	1.1512	19.62231	1656.123	385.0352
3	116.8	962.16	1.3013	314.8756	302960.7	99146.65
5	56.2	551.765	1.3013	135.4638	74744.21	18350.45
7	50.3	455.76	1.0010	91.69422	41790.56	8407.83
8	69.45	312.28	1.0010	133.0094	41536.16	17691.49
9	22.9	173.02	1.0010	37.01045	6403.548	1369.774
10	16.3	147.7	1.0010	25.00843	3693.745	625.4214
11	6.8	60.135	0.8809	8.031553	482.9774	64.50584
12	250	423.055	1.0000	581.8685	246162.4	338570.9
13	278.488	830.285	1.0000	658.9643	547128.2	434233.9
15	7	18.99	0.9105	8.583344	162.9977	73.6738
				SUM	1267616	919113.1

A= 1.3792

Military Mobile Coefficient Calibration, Run 1

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	17.35	88	1.0000	26.84802	2350.947	720.8163
2	30.00	250	1.4456	72.97636	18246.65	5325.55
3	2.31	41	1.3000	3.41511	140.5147	11.66298
4	18.05	418	1.5860	44.57351	18621.92	1986.798
5	3.27	59	1.0000	3.917123	231.4236	15.34385
6	88.63	233	0.7371	129.749	30251.62	16834.8
7	26.24	668	0.8709	37.67034	25156.82	1419.055
9	7.45	190	0.8709	8.818913	1674.712	77.77323
10	6.32	160	0.8709	7.293607	1169.603	53.19671
11	26.81	683	0.8709	38.62373	26363.98	1491.793
				SUM	124208.2	27936.79

A= 4.4460

Military Mobile Coefficient Calibration, Run 2

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	17.35	88	1.0000	26.84802	2350.947	720.8163
2	30.00	250	1.4456	72.97636	18246.65	5325.55
3	2.31	41	1.3000	3.41511	140.5147	11.66298
4	18.05	418	1.5860	44.57351	18621.92	1986.798
6	88.63	233	0.7371	129.749	30251.62	16834.8
7	26.24	668	0.8709	37.67034	25156.82	1419.055
8	32.46	826	0.8709	48.1504	39775.36	2318.461
9	7.45	190	0.8709	8.818913	1674.712	77.77323
11	26.81	683	0.8709	38.62373	26363.98	1491.793
12	58.79	1496	1.2345	135.363	202501.7	18323.15
				SUM	365084.2	48509.85

A= 7.5260

Military Mobile Coefficient Calibration, Run 3

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	17.35	87.565	1.0000	26.84802	2350.947	720.8163
2	30	250.035	1.4456	72.97636	18246.65	5325.55
3	2.311	41.145	1.3000	3.41511	140.5147	11.66298
4	18.052	417.78	1.5860	44.57351	18621.92	1986.798
5	3.268	59.08	1.0000	3.917123	231.4236	15.34385
6	88.633	233.155	0.7371	129.749	30251.62	16834.8
8	32.464	826.065	0.8709	48.1504	39775.36	2318.461
9	7.448	189.9	0.8709	8.818913	1674.712	77.77323
11	26.814	682.585	0.8709	38.62373	26363.98	1491.793
12	58.789	1495.99	1.2345	135.363	202501.7	18323.15
				SUM	340158.8	47106.14

A= 7.2211

Military Mobile Coefficient Calibration, Run 4

Project	SIZE	PM	EM	Q	PM*Q	Q2
1	17.35	88	1.0000	26.84802	2350.947	720.8163
2	30.00	250	1.4456	72.97636	18246.65	5325.55
4	18.05	418	1.5860	44.57351	18621.92	1986.798
5	3.27	59	1.0000	3.917123	231.4236	15.34385
6	88.63	233	0.7371	129.749	30251.62	16834.8
7	26.24	668	0.8709	37.67034	25156.82	1419.055
8	32.46	826	0.8709	48.1504	39775.36	2318.461
10	6.32	160	0.8709	7.293607	1169.603	53.19671
11	26.81	683	0.8709	38.62373	26363.98	1491.793
12	58.79	1496	1.2345	135.363	202501.7	18323.15
				SUM	364670	48488.96

A= 7.5207

Military Mobile Coefficient Calibration, Run 5

Project	SIZE	PM	EM	Q	PM*Q	Q2
2	30.00	250	1.4456	72.97636	18246.65	5325.55
3	2.31	41	1.3000	3.41511	140.5147	11.66298
4	18.05	418	1.5860	44.57351	18621.92	1986.798
5	3.27	59	1.0000	3.917123	231.4236	15.34385
6	88.63	233	0.7371	129.749	30251.62	16834.8
7	26.24	668	0.8709	37.67034	25156.82	1419.055
8	32.46	826	0.8709	48.1504	39775.36	2318.461
10	6.32	160	0.8709	7.293607	1169.603	53.19671
11	26.81	683	0.8709	38.62373	26363.98	1491.793
12	58.79	1496	1.2345	135.363	202501.7	18323.15
				SUM	362459.6	47779.8

A= **7.5860**

Appendix C2. Default and Calibration Estimates

Military Ground—Command & Control, Default and Calibration Effort Estimates

		PM_{UNCAL}			PM_{CAL}				
PROJ	PM_{ACTUAL}	OPTIM	MOST	PESSIM	RUN 1	RUN 2	RUN 3	RUN 4	RUN 5
1	126.60	131.0667	163.8334	204.7918	137.1720	132.1233	174.0312	133.3537	124.6939
2	545.44	423.0504	528.8130	661.0163	442.7568	426.4607	561.7289	430.4322	402.4807
3	721.62	461.7276	577.1594	721.4493	483.2356	465.4496	613.0847	469.7842	439.2772
4	100.23	72.4749	90.5936	113.2420	75.8508	73.0591	96.2326	73.7395	68.9509
5	146.65	66.1713	82.7142	103.3927	69.2537	66.7048	87.8627	67.3260	62.9539
6	339.71	601.8201	752.2751	940.3439	629.8538	606.6715	799.1004	612.3212	572.5581
7	106.56	49.4818	61.8523	77.3154	51.7868	49.8807	65.7023	50.3452	47.0759
8	105.50	59.1936	73.9920	92.4900	61.9509	59.6708	78.5976	60.2265	56.3155
9	301.73	227.7927	284.7408	355.9260	238.4036	229.6289	302.4645	231.7674	216.7168
10	78.07	19.8313	24.7891	30.9863	20.7550	19.9911	26.3321	20.1773	18.8670
11	181.46	122.8448	153.5561	191.9451	128.5672	123.8351	163.1141	124.9884	116.8718
12	176.19	214.7083	268.3854	335.4817	224.7097	216.4391	285.0910	218.4547	204.2686

Military Ground—Signal Processing, Default and Calibration Effort Estimates

		PM_{UNCAL}			PM_{CAL}				
PROJ	PM_{ACTUAL}	OPTIM	MOST	PESSIM	RUN 1	RUN 2	RUN 3	RUN 4	RUN 5
2	174.075	169.9660	212.4575	265.5718	288.9508	245.4100	281.9094	280.3658	282.3863
4	778.59	270.8464	338.5580	423.1975	460.4527	391.0691	449.2320	446.7722	449.9920
5	202.56	95.7046	119.6307	149.5384	162.7027	138.1857	158.7378	157.8686	159.0063
6	293.29	164.3809	205.4761	256.8451	279.4559	237.3459	272.6458	271.1530	273.1071
7	680.475	506.7559	633.4449	791.8061	861.5109	731.6935	840.5167	835.9145	841.9387
8	240.54	155.9980	194.9974	243.7468	265.2045	225.2419	258.7417	257.3250	259.1795
9	278.52	75.7139	94.6424	118.3030	128.7175	109.3216	125.5808	124.8932	125.7933
10	162.47	34.7997	43.4997	54.3746	59.1613	50.2466	57.7196	57.4036	57.8173
11	289.07	221.0066	276.2583	345.3228	375.7225	319.1065	366.5665	364.5594	367.1867
12	200.45	42.4097	53.0121	66.2652	72.0987	61.2344	70.3417	69.9566	70.4607
14	367.14	94.3241	117.9051	147.3814	160.3557	136.1924	156.4480	155.5914	156.7127
15	90.73	75.4057	94.2572	117.8214	128.1936	108.8766	125.0696	124.3848	125.2812
16	152.975	98.1886	122.7357	153.4197	166.9256	141.7723	162.8578	161.9661	163.1333
17	202.56	105.5099	131.8874	164.8592	179.3722	152.3434	175.0011	174.0429	175.2972
18	157.195	32.8639	41.0799	51.3498	55.8703	47.4514	54.5088	54.2103	54.6010
19	114.995	24.5780	30.7225	38.4031	41.7839	35.4876	40.7656	40.5424	40.8346

Ground in Support of Space, Default and Calibration Effort Estimates

		PM_{UNCAL}			PM_{CAL}				
PROJ	PM_{ACTUAL}	OPTIM	MOST	PESSIM	RUN 1	RUN 2	RUN 3	RUN 4	RUN 5
1	64.355	27.25534	34.06918	42.58647	25.13749	16.41856	18.88267	21.61933	19.17886
2	84.4	38.45974	48.07467	60.09334	35.47126	23.16807	26.64514	30.50681	27.0631
3	962.16	617.1562	771.4453	964.3066	569.2006	371.7736	427.5696	489.5371	434.2764
4	121.325	37.39238	46.74047	58.42559	34.48684	22.52509	25.90567	29.66017	26.31202
5	551.765	265.5091	331.8864	414.858	244.878	159.9422	183.9464	210.6056	186.8317
6	504.29	171.5067	214.3834	267.9793	158.18	103.3153	118.8209	136.0416	120.6847
7	455.76	179.7207	224.6508	280.8136	165.7556	108.2634	124.5116	142.557	126.4647
8	312.28	260.6983	325.8729	407.3411	240.441	157.0441	180.6134	206.7896	183.4465
9	173.02	72.54049	90.67561	113.3445	66.90379	43.69824	50.25649	57.54015	51.04482
10	147.7	49.01652	61.27065	76.58831	45.20773	29.52745	33.95894	38.8806	34.49162
11	60.135	15.74184	19.6773	24.59663	14.51864	9.482855	10.90605	12.48666	11.07712
12	423.055	1140.462	1425.578	1781.972	1051.844	687.0121	790.1192	904.6309	802.513
13	830.285	1291.57	1614.463	2018.078	1191.21	778.0391	894.8076	1024.492	908.8435
14	60.135	89.44083	111.801	139.7513	82.49091	53.87897	61.96516	70.94575	62.93714
15	18.99	16.82335	21.02919	26.28649	15.51611	10.13435	11.65532	13.34453	11.83815

Military Mobile, Default and Calibration Effort Estimates

		PM_{UNCAL}			PM_{CAL} MOST LIKELY				
PROJ	PM_{ACTUAL}	OPTIM	MOST	PESSIM	RUN 1	RUN 2	RUN3	RUN 4	RUN 5
1	87.565	52.62212	65.77765	82.22207	119.3663	202.0582	193.8723	201.9159	203.6691
2	250.035	143.0337	178.7921	223.4901	324.4529	549.2201	526.9696	548.8333	553.5987
3	41.145	6.693615	8.367019	10.45877	15.18358	25.70212	24.66085	25.68402	25.90702
4	417.78	87.36409	109.2051	136.5064	198.1738	335.4603	321.8698	335.224	338.1347
5	59.08	7.677561	9.596951	11.99619	17.41553	29.48027	28.28594	29.45951	29.71529
6	233.155	254.308	317.885	397.3563	576.864	976.4909	936.9304	975.8032	984.2758
7	667.815	73.83387	92.29233	115.3654	167.4823	283.507	272.0213	283.3073	285.7672
8	826.065	94.37478	117.9685	147.4606	214.0767	362.3799	347.6988	362.1247	365.2689
9	189.9	17.28507	21.60634	27.00792	39.20889	66.37114	63.68225	66.3244	66.90028
10	160.36	14.29547	17.86934	22.33667	32.42738	54.89169	52.66787	54.85303	55.3293
11	682.585	75.70252	94.62815	118.2852	171.7211	290.6822	278.9058	290.4775	292.9996
12	1495.99	265.3115	331.6394	414.5492	601.824	1018.742	977.4699	1018.025	1026.864

Appendix D. Regression

Military Ground—C^2, SAS® Regression Output

Adjusted R-square	R-square In		Variables in Model
0.87293554	**0.91914080**	**4**	**PMDEF DEFSQ SQRTDEF RECDEF**
0.78468742	0.84340903	3	PMDEF DEFSQ SQRTDEF
0.70160157	0.78298296	3	PMDEF DEFSQ RECDEF
0.69176052	0.74780406	2	PMDEF DEFSQ
0.65256773	0.68415248	1	SQRTDEF
0.64645144	0.74287377	3	DEFSQ SQRTDEF RECDEF
0.63038824	0.66398931	1	PMDEF
0.62693054	0.69476135	2	SQRTDEF RECDEF
0.62391824	0.69229674	2	DEFSQ SQRTDEF
0.61399743	0.68417972	2	PMDEF SQRTDEF
0.60632887	0.71369372	3	PMDEF SQRTDEF RECDEF
0.59108862	0.66543614	2	PMDEF RECDEF
0.47266383	0.52060349	1	DEFSQ
0.47098120	0.56716644	2	DEFSQ RECDEF
0.23077322	0.30070292	1	RECDEF

Military Ground—Signal Processing, SAS® Regression Output

Adjusted R-square	R-square	In	Variables in Model
0.59601279	0.64987775	2	SQRTDEF RECDEF
0.58613816	**0.61372895**	**1**	**SQRTDEF**
0.57952700	0.60755853	1	PMDEF
0.57212931	0.62917873	2	DEFSQ SQRTDEF
0.57021060	0.65616848	3	PMDEF SQRTDEF RECDEF
0.56538791	0.65231033	3	DEFSQ SQRTDEF RECDEF
0.56147557	0.61994549	2	PMDEF SQRTDEF
0.54898356	0.60911909	2	PMDEF DEFSQ
0.54749981	0.60783317	2	PMDEF RECDEF
0.53704643	0.62963714	3	PMDEF DEFSQ SQRTDEF
0.53154945	0.65646960	4	PMDEF DEFSQ SQRTDEF RECDEF
0.52419193	0.58763301	2	DEFSQ RECDEF
0.51160579	0.60928463	3	PMDEF DEFSQ RECDEF
0.50983988	0.54251722	1	DEFSQ
0.22354035	0.27530433	1	RECDEF

Ground in Support of Space, SAS® Regression Output

Adjusted R-square	R-square	In	Variables in Model
0.75209020	0.78750589	2	DEFSQ SQRTDEF
0.74367751	0.78029501	2	PMDEF DEFSQ
0.73891522	0.79486196	3	PMDEF SQRTDEF RECDEF
0.73846022	0.79450446	3	DEFSQ SQRTDEF RECDEF
0.73740679	0.77492010	2	PMDEF SQRTDEF
0.73220806	0.78959205	3	PMDEF DEFSQ SQRTDEF
0.72310409	0.78243893	3	PMDEF DEFSQ RECDEF
0.71414448	0.79581749	4	PMDEF DEFSQ SQRTDEF RECDEF
0.64288640	0.69390263	2	SQRTDEF RECDEF
0.63651597	0.66247912	1	SQRTDEF
0.58389344	0.64333724	2	PMDEF RECDEF
0.51488551	0.58418758	2	DEFSQ RECDEF
0.48046906	0.51757841	1	PMDEF
0.43834860	0.47846655	1	RECDEF
0.29072125	0.34138402	1	DEFSQ

Military Mobile, SAS® Regression Output

Adjusted R-square	R-square	In	Variables in Model
0.33808129	0.39825572	1	SQRTDEF
0.29957751	0.36325228	1	PMDEF
0.27544552	0.40718270	2	DEFSQ RECDEF
0.27079951	0.40338142	2	PMDEF RECDEF
0.27066989	0.40327537	2	SQRTDEF RECDEF
0.26740274	0.40060224	2	PMDEF SQRTDEF
0.26474402	0.39842693	2	DEFSQ SQRTDEF
0.25357884	0.45714825	3	PMDEF DEFSQ SQRTDEF
0.24060483	0.37867668	2	PMDEF DEFSQ
0.22939839	0.29945308	1	DEFSQ
0.22749087	0.29771897	1	RECDEF
0.21457971	0.50018709	4	PMDEF DEFSQ SQRTDEF RECDEF
0.18676953	0.40855966	3	DEFSQ SQRTDEF RECDEF
0.18488118	0.40718631	3	PMDEF DEFSQ RECDEF
0.18043372	0.40395180	3	PMDEF SQRTDEF RECDEF

Military Ground—C^2, Final Regression Model

PROJECT	ACTUAL	DEFAULT	DEF SQ	SQRT DEF	1/DEF	PMregress	MRE
1	126.60	163.8334	26841	12.7997	0.0061	108.51909	0.1428192
2	545.44	528.8130	279643	22.9959	0.0019	610.28473	0.1188954
3	721.62	577.1594	333113	24.0241	0.0017	610.02036	0.1546515
4	100.23	90.5936	8207	9.5181	0.0110	105.93192	0.0569411
5	146.65	82.7142	6842	9.0947	0.0121	115.51987	0.2122481
6	339.71	752.2751	565918	27.4276	0.0013	368.96595	0.0861204
7	106.56	61.8523	3826	7.8646	0.0162	148.57542	0.3943542
8	105.50	73.9920	5475	8.6019	0.0135	128.33792	0.2164732
9	301.73	284.7408	81077	16.8743	0.0035	299.54572	0.0072392
10	78.07	24.7891	614	4.9789	0.0403	69.966335	0.1038
11	181.46	153.5561	23579	12.3918	0.0065	99.891216	0.4495139
12	176.19	268.3854	72031	16.3825	0.0037	269.14691	0.5276381
						MMRE	**0.2058912**
						PRED(.25)	**0.75**

Military Ground—Signal Processing, Final Regression Model

PROJECT	ACTUAL	DEFAULT	PMregress	MRE
2	174.075	212.45745	318.03242	0.826985
4	778.59	338.55803	446.71995	0.4262449
5	202.56	119.63074	223.30137	0.1023962
6	293.29	205.4761	310.90784	0.0600697
7	680.475	633.44487	747.6564	0.0987272
8	240.54	194.99744	300.21421	0.2480844
9	278.52	94.642396	197.80038	0.2898162
10	162.47	43.499666	145.60846	0.1037825
11	289.07	276.25825	383.14209	0.3254301
12	200.45	53.012145	155.31609	0.2251629
14	367.14	117.90509	221.54031	0.3965781
15	90.73	94.257151	197.40724	1.1757659
16	152.975	122.73574	226.47006	0.4804384
17	202.56	131.88738	235.80945	0.1641462
18	157.195	41.079861	143.13901	0.0894175
19	114.995	30.722501	132.56917	0.1528255
			MMRE	**0.3228669**
			Pred(.25)	**0.5625**

Ground in Support of Space, Final Regression Model

PROJECT	ACTUAL	DEFAULT	LN ACTUAL	LN DEFAULT	$PM_{regress}$	MRE
1	64.355	34.06918	4.164415	3.528393	69.44073	0.079026
2	84.4	48.07467	4.435567	3.872755	88.61032	0.049885
3	962.16	771.4453	6.869181	6.648266	632.0937	0.343047
4	121.325	46.74047	4.798473	3.84461	86.86233	0.284053
5	551.765	331.8864	6.313122	5.804793	347.9092	0.369461
6	504.29	214.3834	6.223151	5.367766	255.3332	0.493678
7	455.76	224.6508	6.121966	5.414547	263.9305	0.4209
8	312.28	325.8729	5.7439	5.786507	343.4348	0.099766
9	173.02	90.67561	5.153407	4.507288	138.8557	0.197459
10	147.7	61.27065	4.995183	4.115301	105.2088	0.287686
11	60.135	19.6773	4.096592	2.979466	47.08182	0.217065
12	423.055	1425.578	6.047502	7.262332	976.2675	1.307661
13	830.285	1614.463	6.721769	7.386757	1066.159	0.284087
14	60.135	111.801	4.096592	4.716721	161.0465	1.678083
15	18.99	21.02919	2.943913	3.045912	49.34932	1.5987
					MMRE	**0.514037**
					Pred(.25)	**0.333333**

Military Mobile, Final Regression Model

PROJECT	ACTUAL	DEFAULT	LN ACTUAL	LN DEFAULT	$PM_{regress}$	MRE
1	87.565	65.777655	4.4724	4.1863	264.50783	2.0207027
2	250.035	178.79209	5.5216	5.1862	513.8515	1.0551183
3	41.145	8.3670194	3.7171	2.1243	67.256	0.6346093
4	417.78	109.20511	6.0350	4.6932	370.38215	0.1134517
5	59.08	9.5969512	4.0789	2.2614	73.669303	0.2469415
6	233.155	317.88501	5.4517	5.7617	753.02761	2.2297296
7	667.815	92.292334	6.5040	4.5250	331.22211	0.5040212
8	826.065	117.96847	6.7167	4.7704	389.86361	0.5280473
9	189.9	21.606337	5.2465	3.0730	126.28377	0.3349986
10	160.36	17.869338	5.0774	2.8831	111.32106	0.3058053
11	682.585	94.628149	6.5259	4.5500	336.76577	0.5066317
12	1495.99	331.63939	7.3105	5.8040	774.51127	0.4822751
					MMRE	**0.746861**
					PRED(.25)	**0.166667**

Shapiro-Wilk Test Results for Final Regression Run

Application	W Statistic	P-value	Confidence Level
Mil Grd—C^2	0.972738	0.891773	90%
Mil Grd—Signal Processing	0.966565	0.767314	90%
Grd in Support of Space	0.941697	0.421288	90%
Mil Mobile	0.973701	0.902012	90%

Note: For Mil Grd—Signal Processing, eliminated Project 4 to obtain given results.
For Grd in Support of Space, eliminated Project 12 to obtain given results.

Equal Variance Test Results for Final Regression Run

Application	# of data points	Degrees of Freedom	Calculated F value	Table F value	Equal Variance
Mil Grd—C^2	6	1	14.41	39.86	yes
Mil Grd—Signal Processing	8	6	1.55	3.05	yes
Grd in Support of Space	7	5	0.37	3.45	yes
Mil Mobile	6	4	1.22	4.11	yes

Note: For Mil Grd—Signal Processing, when the original data set was sorted based on actual effort, the middle data point was eliminated to keep the data sets equal in size.
For Grd in Support of Space, note the reduction in the variance as the actual effort increased.

References

Apgar, H., D. Galorath, R. Maness, and S. Stukes. Application Oriented Software Data Collection: Software Model Calibration Report. (TR-9007/49-1) Oxnard CA: Managment Consulting and Research Corporation, 15 March 1991.

Bersoff, E. "Elements of Software Configuration Management," IEEE, SE-10: 79-87 (January 1984).

Boehm, B. Presentation on COCOMO II. Air Force Cost Analysis Agency, Crystal City VA, 11 April 1997.

Boehm, B. Personal Interview. Director to the Center for Software Engineering at the University of Southern California. Air Force Cost Analysis Agency, Crystal City VA, 11 April 1997.

Boehm, B. Software Engineering Economics. Englewood Cliffs NJ: Prentice-Hall, 1981.

Boehm, B., B. Clark, E. Horowitz, C. Westland, R. Madachy, and R. Selby. "The COCOMO 2.0 Software Cost Estimation Model," American Programmer, 9: 2-17 (July 1996).

Brabson, G. D. "Can We Afford the DOD Acquisition Actions?," Concepts, 5: 45-55 (January 1982).

Carnegie Mellon University, & Software Engineering Institute. The Capability Maturity Model: Guidelines for Improving the Software Process. Addison-Wesley Publishing Company, Reading MA, 1995.

Clark, B. Personal Interview. Doctorate Student at the University of Southern California. Williamsburg VA, 12 February 1997.

COCOMO II.1997.0. Version 0, IBM, one 1.4 Mb disk. Computer software. University of Southern California, Los Angeles CA, 1997.

Conover, W. J. Practical Nonparametric Statistics. New York NY: John Wiley & Sons, 1980.

Conte, S. D., H. E. Dunsmore and V.Y. Shen. Software Engineering Metrics and Models. Menlo Park CA: The Benjamin/Cummings Publishing Company, Inc., 1986.

Crosby, Philip B. Quality is Free. New York NY: McGraw-Hill, Inc., 1979.

D'Agostino, R. B., and M. A. Stephens. Goodness-of-Fit Techniques. New York NY: Marcel Dekker, Inc., 1986.

Daly, Bryan A. A Comparison of Software Schedule Estimators. MS Thesis, AFIT/GCA/LSQ/90S-1. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1990 (AD-A229532).

D'Angelo, Anthony. Class Notes, AMGT 602. School of Logistics and Acquisition Management, Air Force Institute of Technology, Wright-Patterson AFB OH, Summer 1997.

Dechoretz, J. and S. Stukes. Software Estimating Model Improvement Project REVIC Recalibration. (TR-9359/51-1) Oxnard CA: Management Consulting and Research Corporation, 18 March 1994.

DeMarco, Tom. Controlling Software Projects. Englewood Cliffs NJ: Prentice-Hall Inc., 1982.

DeMarco, Tom. Why Does Software Cost So Much?. New York NY: Dorset House Publishing Co., Inc., 1995.

Department of Defense. 5000.2-R, Mandatory Procedures for Major Defense Acquisition Programs and Major Automated Information System Acquisition Programs. 15 March 1996.

Department of the Air Force. Guidelines for Successful Acquisition and Management of Software-Intensive Systems: Weapon Systems, Command and Control Systems, Management Information Systems. Volume 1, Version 2. Hill AFB UT: Ogden Air Logistics Center, June 1996. (Special Note: This publication is separated into several sections with repeated pagination, therefore the section as well as the page number(s) have been referenced)

Dolkas, James B., Michael W. Evans, & Pamela Piazza. Principles of Productive Software Management. New York NY: John Wiley and Sons, Inc., 1983.

Donald, B. Personal Interview. Air Force Cost Analysis Agency, Crystal City VA, 11 April 1997.

Ferens, Daniel V. Class Handouts, Cost 677 Note-Taking Devices. School of Logistics and Acquisition Management, Air Force Institute of Technology, Wright-Patterson AFB OH, Winter 1997.

Ferens, Daniel V. Professor, Air Force Institute of Technology School of Logistics and Acquisition Management. "Results of the 'Septuagint' Study." Presentation to Space Systems Cost Analysis Group Meeting. Tysons Corner VA 22 October 1996.

Ferens, Daniel V. "Software Cost Estimation in the DOD Environment," American Programmer, 9: 28-34 (July 1996).

Galonsky, James C. Calibration of the PRICE S Software Cost Model. MS Thesis, AFIT/GCA/LAS/95S-1. School of Logistics and Acquisition Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1995 (AD-A301337).

Galorath Associates, Inc. SEER-SEM User's Manual, Release 4.5. El Segundo CA: Galorath Associates, Inc., 1996.

Gansler, Jacques S. Affording Defense. Cambridge MA: MIT Press, 1989.

Goel, Amrit L. "Software Reliability Models," IEEE Transitions on Software Engineering, SE-10: 1411-1423.

IIT Research Institute. Test Case Study: Evaluating the Cost of Ada Software Development. Langham MD: IIT Research Institute, 1989.

Jensen, Randall W. "A New Perspective in Software Schedule and Estimation," Article published by Software Engineering, Inc., Brigham City UT. 1996.

Jones, Capers. "How Software Estimation Tools Work," American Programmer, 9: 18-27 (July 1996).

Jones, Capers. Programming Productivity. New York NY: McGraw-Hill, Inc., 1986.

Karolak, Dale W. Software Engineering Risk Management. Los Alamitos CA: IEEE Computer Society Press, 1996.

Kemerer, Chris F. "An Empirical Validation of Software Cost Estimation Models," Communications of the ACM, 30: 416-429 (May 1987).

Kile, Ray. REVIC Software Cost Estimating Model User's Manual. Unpublished. 1991.

Kressin, Robert K. Calibration of SLIM to the Air Force Space and Missile Systems Center Software Database. MS Thesis, AFIT/GCA/LAS/95S-6. School of Logistics and Acquisition Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1995 (AD-A301603).

Lederer, Albert L. and J. Prasad. "Causes of Inaccurate Software Development Cost Estimates," Journal of Systems Software, 31: 125-134 (November 1995).

Lokan, Christopher J. "Early Size Prediction for C and Pascal Programs," Journal of Systems Software, 32: 65-72 (January 1996).

Magee, Robert P. Advanced Managerial Accounting. New York NY: Harper and Row, 1986.

Marzo, David. Calibration of SAGE to the Air Force Space and Missile Systems Center Software Database and Electronic Systems Center Database. MS Thesis, AFIT/GCA/LAS/97S-6. School of Logistics and Acquisition Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, in progress, to be published later.

Management and Consulting Research, Inc. and Cost Management Systems, Inc. SMC SWDB Space and Missile Systems Center Software Database User's Manual Version 2.1. El Segundo CA: SMC/FMC, 1995.

Matson, J., B. Barrett, and J. Mellichamp. "Software Development Cost Estimation Using Function Points," IEEE Transactions on Software Engineering, 20: 275-287 (April 1994).

Meldenhall, W., D. Wackerly, and R. Scheaffer. Mathematical Statistics with Applications (Fourth Edition). Belmont CA: Duxbury Press, 1990.

Mertes, Karen R. Calibration of the CHECKPOINT Model to the Space and Missile Systems Center (SMC) Software Database. MS Thesis, AFIT/GCA/LAS/96S-11. School of Logistics and Acquisition Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1996 (AD-A319518).

Mosemann, Lloyd K. II, Deputy Assistant Secretary of the Air Force for Communications, Computers, and Support Systems. Speech to the Software Technology Conference, Salt Lake City UT, 1994.

Object Management Group. OMG. 1997. Http://www.omg.org.

Ourada, Gerald L. Software Cost Estimating Models: A Calibration, Validation and Comparison. MS Thesis, AFIT/GSS/LSY/91D-11. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1991 (AD-A246677).

Rathmann, Kolin D. Calibration and Evaluation of SEER-SEM for the Air Force Space and Missile System Center. MS Thesis, AFIT/GCA/LAS/95S-9. School of Logistics and Acquisition Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1995 (AD-A300703).

Rational Software Corporation. Unified Modeling Language Resource Center. January 13, 1997. Http://www.rational.com/uml/index.html.

Shrum, Tom. Calibration of CHECKPOINT to the Electronic Systems Center Database. MS Thesis, AFIT/GCA/LAS/97S-7. School of Logistics and Acquisition Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, in progress, to be published later.

SMC SWDB. Version 2.1, IBM, five 1.4 Mb disks. Computer software. SMC/FMC, El Segundo CA, 1995.

Softkey, Sheldon D. The ABC's of Developing Software. Menlo Park CA: The ABC Press of Silicon Valley, 1983.

Software Productivity Research, Inc. CHECKPOINT for Windows User's Guide, Release 2.2.2. Burlington MA: Software Productivity Research, Inc., 1993.

Southwell, Steven V. Calibration of the SOFTCOST-R Software Cost Model to the Space and Missile Systems Center (SMC) Software Database (SWDB). MS Thesis, AFIT/GSM/LAS/96S-6. School of Logistics and Acquisition Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1996 (AD-A319050).

Statgraphics Plus for Windows. Version 2, IBM, five 1.4 Mb disks. Computer software. Manugistics, Inc., Rockville MD, 1995.

Stukes, Sherry. Space and Missile Systems Center Software Database Collection Effort Final Report. Contract F04701-95-D-0003, Task 003. Oxnard CA: Management Consulting and Research, 15 November 1995.

Stukes, Sherry and David Patterson. Presentation of the Space and Missile Systems Center Software Database. Air Force Institute of Technology, School of Logistics and Acquisition Management, Wright-Patterson AFB OH, 12 November 1996.

Stutzke, Richard D. "Software Estimating Technology: A Survey," Crosstalk: 17-21 (May 1996).

Thibodeau, Robert. An Evaluation of Software Cost Estimating Models. Huntsville AL: General Research Corporation, 1981.

University of Southern California. COCOMO II Model Definition Manual, Version 1.4. Los Angeles CA: 1997.

University of Southern California. COCOMO II.1997 Reference Manual. Los Angeles CA, 1997.

Vegas, C. D. Calibration of Software Architecture Sizing and Estimating Tool. MS Thesis, AFIT/GCA/LAS/95S-11. School of Logistics and Acquisition Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1995 (AD-A301376).

Weber, Betty G. A Calibration of the REVIC Software Cost Estimating Model. MS Thesis, AFIT/GCA/LAS/95S-13. School of Logistics and Acquisition Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1995 (AD-A300694).

Vita

Wayne A. Bernheisel [redacted] graduated from Sylvania Southview High School in 1983. In August 1985, he graduated from the University of Cincinnati with an Associate of Science in Electrical Engineering Technology. His first full-time employment was with Unisys Corporation, United States Parts Recovery Operations in Holland, Ohio, repairing various computer systems. While working, he attended night school at the University of Toledo where he graduated in 1991 with a Bachelor of Science in Business. Upon graduation, he worked as a Financial Planner until being accepted and entering the Officer Training School at Maxwell Air Force Base, Alabama, in May 1994.

Lieutenant Bernheisel's first Air Force assignment was at the 20th Bomb Squadron, Barksdale AFB, Louisiana, from August 1994 to January 1996. While there, he served as the Squadron Section Commander. Prior to entering AFIT, he was given the opportunity to work in the 2d Bomb Wing Command Post to become familiar with its operations and broaden his work experience and operations knowledge.

Lieutenant Bernheisel entered AFIT's Graduate Cost Analysis program in May 1996. Upon graduation, he anticipates a follow-on assignment to the F-16 Systems Program Office at Wright-Patterson AFB, Ohio.

[redacted]

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 074-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503			
1. AGENCY USE ONLY (Leave blank)	**2. REPORT DATE** September 1997	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE CALIBRATION AND VALIDATION OF THE COCOMO II.1997.0 COST/SCHEDULE ESTIMATING MODEL TO THE SPACE AND MISSILE SYSTEMS CENTER DATABASE			**5. FUNDING NUMBERS**
6. AUTHOR(S) Wayne A. Bernheisel, First Lieutenant, USAF			
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology 2950 P Street WPAFB OH 45433-7765			**8. PERFORMING ORGANIZATION REPORT NUMBER** AFIT/GCA/LAS/97S-1
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) SMC/FMC 2430 East El Segundo Boulevard, Suite 2010 Los Angeles, CA 90245-4687			**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution unlimited			**12b. DISTRIBUTION CODE**

13. ABSTRACT ***(Maximum 200 Words)***

The goal of this study was to determine the accuracy of COCOMO II.1997.0, a software cost and schedule estimating model, using Magnitude of Relative Error, Mean Magnitude of Relative Error, Relative Root Mean Square, and a 25 percent Prediction Level. Effort estimates were completed using the model in default and in calibrated mode. Calibration was accomplished by dividing four stratified data sets into two random validation and calibration data sets using five times resampling. The accuracy results were poor; the best having an accuracy of only .3332 within 40 percent of the time in calibrated mode. It was found that homogeneous data is the key to producing the best results, and the model typically underestimates. The second part of this thesis was to try and improve upon the default mode estimates. This was accomplished by regressing the model estimates to the actual effort. Each original regression equation was transformed and tested for normality, equal variance, and significance. Overall, the results were promising; regression improved the accuracy in three of the four cases, the best having an accuracy of .2059 within 75 percent of the time.

14. Subject Terms Cost Analysis, Cost Estimates, Cost Models, Validation, Software, Models, Cost Overruns, Sampling, Software Engineering			**15. NUMBER OF PAGES** 142 **16. PRICE CODE**
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	**18. SECURITY CLASSIFICATION OF THIS PAGE** UNCLASSIFIED	**19. SECURITY CLASSIFICATION OF ABSTRACT** UNCLASSIFIED	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

AFIT Control Number AFIT/GCA/LAS/97S-1

AFIT RESEARCH ASSESSMENT

The purpose of this questionnaire is to determine the potential for current and future applications of AFIT thesis research. **Please return completed questionnaire** to: AIR FORCE INSTITUTE OF TECHNOLOGY/LAC, 2950 P STREET, WRIGHT-PATTERSON AFB OH 45433-7765. Your response is **important.** Thank you.

1. Did this research contribute to a current research project? a. Yes b. No

2. Do you believe this research topic is significant enough that it would have been researched (or contracted) by your organization or another agency if AFIT had not researched it?
a. Yes b. No

3. **Please estimate** what this research would have cost in terms of manpower and dollars if it had been accomplished under contract or if it had been done in-house.

Man Years____________ $____________

4. Whether or not you were able to establish an equivalent value for this research (in Question 3), what is your estimate of its significance?

a. Highly Significant b. Significant c. Slightly Significant d. Of No Significance

5. Comments (Please feel free to use a separate sheet for more detailed answers and include it with this form):

Name and Grade

Organization

Position or Title

Address